

# Contents

Contents.....	1
Outline.....	3
What is IS01? .....	3
General functions .....	3
Hardware functions .....	3
Software functions.....	3
System requirements .....	4
Standard package of the programmer.....	4
Hardware installation .....	5
Connection of ISP cable with the target system.....	5
Description of connection port.....	5
Definition of several common interfaces .....	6
Examples for connection of target system .....	7
ISP requirements for target system .....	8
Attentions on pin connection.....	8
Precautions for power: .....	10
Precautions for the design with reset generator .....	10
Examples of common problems.....	12
Other matters needing attention .....	13
Examples for connection of several common serial communication protocols.....	13
M25P64 .....	13
24C04 .....	14
S9S08AW16.....	14
Use ATE interface to control the programmer.....	14
Time sequence of ATE interface.....	15
Installation of software.....	15
Quick start .....	25
User interface of application software .....	25
Steps for chip burning .....	26
Hardware preparation.....	26
Select device.....	26
Load data into buffer .....	27
Set options .....	27
Programming.....	28
Detailed description of functions .....	28
Browse menu.....	28
Load file .....	28
Save file.....	30
Exit .....	30
Load project.....	30
Save project.....	31

Project library management .....	31
SD card operation.....	32
Select chip .....	32
XPlayer.....	34
Buffer .....	34
Checksum.....	38
Device information.....	39
Device Configuration Word.....	39
Edit automatic processing command.....	39
Operation options .....	40
Module management .....	43
Factory mode.....	44
Print Report .....	45
Tool bar .....	45
Main toolbar .....	45
Operation toolbar.....	46
Information bar.....	46
Device information.....	46
Buffer information.....	46
File information.....	46
Statistical Information Bar .....	47
Log window.....	47
LED status display .....	47
Progress bar .....	47
Operation information bar .....	47
Offline operation .....	48
Offline introduction.....	48
Outline.....	48
Menu description.....	49
Appendices.....	40

# Outline

## What is IS01?

IS01 is a professional multifunction serial programmer with small volume, fast programming speed, good stability and high cost performance; it supports almost all serial programmable devices and is very suitable for In-System or In-Application applications. It can also support IC offline programming with certain IC sockets .

### General functions

- Very fast programming speed, which is selectable to adapt to cable length and load feature of user board.
- Supports online (USB2.0 high-speed) and offline mode simultaneously.
- Supports most ISP protocol, e.g. I2C, SPI, UART, BDM, MW, JTAG, CAN, RS232 etc.
- Have standard ATE interface to facilitate operation and control of external ATE device.
- Have overcurrent protection function to protect your equipment comprehensively.
- Apply to many fields attributing to stable and reliable design, for example, R & D, mass production as well as in-system programming etc.

### Hardware functions

- Provides interfaces for I2C, SPI, UART, BDM, MW, JTAG, RS232, CAN etc.
- Provides VDD(0.5A) output line to supply light-load target system with power to directly (generally not recommended) and VPP (0.2A).
- Long cable drive circuit has ESD protection, I/O level can be adjusted by software in the range of 1.5V~4V.
- Online software operates the programmer and manages offline project via USB (2.0 high-speed).
- In offline state, the operations are shown through LCD screen, 6-key pad and LED status, and the project file is stored using SD card (max. 4GB).
- ATE interface provides the possibility of operating and controlling programmer through external signals. Among them, there are 6 project selection lines (SEL0~SEL5), 2 command input lines (START and STOP) as well as 3 status output lines (PASS, FAIL and BUSY). External signals are isolated from the system all through optocouplers.
- 12VDC/2A system external power input.

### Software functions

- Support a wide range of devices, can support almost all the chips with ISP function (Attention: not all algorithms for programming are free of charge).

- Very fast programming rate, which is selectable (three gears as high, medium and low) to adapt to different cable length and load feature of user board.
- Support dynamic buffer function (each chip burns different contents, the application example includes serial number, MAC address, sensor calibration etc.), the interface is of open type.
- SD card supports FAT32, FAT16 etc.; can be managed using card reader.
- Provide good software interface (DLL, not free of charge) to facilitate third-party system integration. Typical application such as ICT (online tester).
- Operate up to 12 programmers through one USB HUB. Chips and files can be different.

## **System requirements**

Minimum requirement for system is as follows: Pentium or above compatible computer, desktop or portable computer; at least there is a universal serial bus interface to USB2.0 standard; Windows XP/Vista operating system; CD-ROM; hard disc with at least 1GB free space.

## **Standard package of the programmer**

The standard package includes the following items:

- One programmer ;
- One ISP cable, type A (supports interface other than CAN and RS232)
- One connecting cable for USB interface
- One power adapter (output DC12V)
- Installation software, one CD-ROM disc
- One sheet of user registration form.

Options:

- ISP cable, type B (supports CAN and RS232 interface)
- Adapter kit for offline programming. If you want to perform offline programming on the device like a universal programmer, you only need connect ISP cable to the kit board and insert the chip into the kit socket. The adapter kit consists of the upper and lower boards, the upper board is the adapter for SP 5000, the user can select it according to the chip package. The lower board is the jumper adapter.

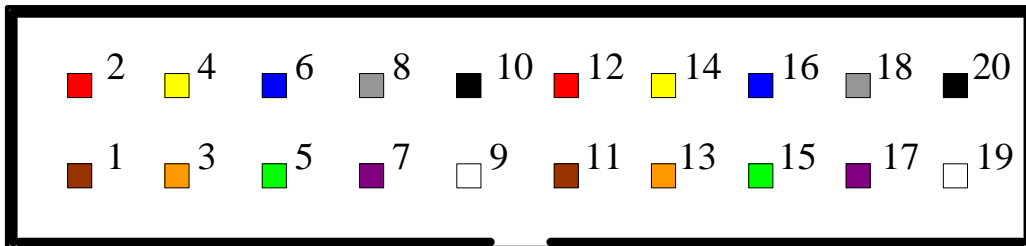
## Hardware installation

### Connection of ISP cable with the target system

#### Description of connection port

ISP connection port contains the signal ends required for programming of target system. The connection port is divided into group A and group B:

Port A is a 10X2 port. The programming interfaces required for most serial protocols are contained in this port, e.g. SPI, I2C, JTAG, UART, BDM etc., the pins of this port are defined as follows:



Pin #	Signal Name	Description
1	GND	Ground
2	GND	Ground
3	DIO0	Digital input/output 0
4	DIO1	Digital input 1
5	DIO2	Digital output 2
6	DIO3	Digital input 3
7	DIO4	Digital output 4
8	DIO5	Digital output 5
9	DIO6	Digital input/output 6
10	DIO7	Digital output 7
11	SCL	Special function Pin for I2C SCL
12	SDA	Special function Pin for I2C SDA
13	EVOL	target system power on detection(input)
14	Reserved1	Reserved digital input 1
15	Reserved2	Reserved digital output 1
16	Reserved3	Reserved digital output 1
17	VPROG1	Programmable voltage 1(max 5.5V, 500mA)
18	VPROG1	Programmable voltage 1(max 5.5V, 500mA)
19	VPROG0	Programmable voltage 0(max 18V, 250mA)
20	VPROG0	Programmable voltage 0(max 18V, 250mA)

#### Notes:

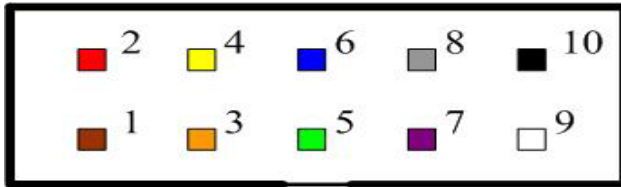
1. IS01 has a working mode called “mass production”, that is, once the programmer detects that the target board gets ready and ISP signal arrives, it immediately starts the functional operation defined by AUTO without keyboard operation. TPIN and TPOUT in

the port are provided for this purpose. TPOUT supplies high level, and TPIN is level detection input line. Short-circuit them on the target board, once ISP signal is applied to the test point, TPIN immediately detects high level, and AUTO command sequence can be started.

2. Reserved1-2 is only used when the multi-object system is programming. On some target board there are several identical I2C chips that need to be burned, and the chip that needs to be burned currently can be selected through address line, then, Reserved1-2 is connected to A0 and A2, respectively.

+

Port B is a 5X2 port. CAN and RS-232 come out from this port. The pins are defined as below:



Pin #	Signal Name	Description
1	EVOL	target system power on detection(input)
2	RX	RX(input, RS-232 levels)
3	TX	TX(output, RS-232 levels)
4	VPROG0	Programmable voltage 0(max 18V, 250mA)
5	GND	Ground
6	VPROG1	Programmable voltage 1(max 18V, 250mA)
7	RTS	RTS(input, RS-232 levels)
8	CTS	CTS(input, RS-232 levels)
9	CANH	CAN Bus difference signal for high
10	CANL	CAN Bus difference signal for low

### Definition of several common interfaces

In order to facilitate use by customer, we have defined several common interfaces corresponding to ISP port (this definition only contains communication protocol pins, for the specific wiring method, please refer to the hint on the software of host computer), the pin correspondence table is as follows:

Pin #	Signal Name	IO Type	SPI	I2C	JTAG	BDM	UART
1	GND						
2	GND						
3	DIO0	Input/Output					
4	DIO1	Input					CTS
5	DIO2	Output	SPIMOSI		TDO		TXD
6	DIO3	Input	SPIMISO		TDI		RXD
7	DIO4	Output	SPICLK	SCL	TCK		RTS
8	DIO5	Output			TRST	RESET	
9	DIO6	Input/Output		SDA		BDM	
10	DIO7	Output			TMS		
11	SCL						
12	SDA						
13	EVOL						
14	Reserved1						
15	Reserved2						
16	Reserved3						
17	VPROG1						
18	VPROG1						
19	VPROG0						
20	VPROG0						

Note: SDA and SCL are reserved for test, and generally are not provided for user.

### Examples for connection of target system

Before programming the target device, you must be familiar with the target device, learn the function of each pin of the device; read on the instructions on programming of this device with ISP programmer carefully; know the hardware programming interfaces on the target board; and confirm compliance of target board design with ISP requirements (see the section “ISP requirements for the target system” for detail).

The programming operation steps are as follows:

1. Search the type of the chip to be operated in ISP programming software and view the operation hint.
2. Exactly connect the signal lines (including GND) for the corresponding interfaces of ISP programmer with the target board.
3. If the above “mass production” function is to be used, TPIN and TPOUT signal lines should also be connected.
4. It is recommended that power of target board is supplied independently, especially for the target board with high power consumption and with multiple power systems. If it is confirmed that the power is supplied by programmer ISP cable, then, VCC (sometimes also need Vpp) should be connected.
5. The independent power supplier of target board switches on the power.
6. Operation functions for running the programmer chip.

Take AT Mega48 as an example, according to the operation hint, we should make line connection according to the table below:

ISP Programmer		Target system
Pin #	Signal Name	Connection
1	GND	GND
2	GND	GND
3	DIO0	No connect
4	DIO1	No connect
5	DIO2	MOSI
6	DIO3	MISO
7	DIO4	SCK
8	DIO5	RESET
9	DIO6	No connect
10	DIO7	No connect
11	SCL	No connect
12	SDA	No connect
13	EVOL	(1.8-12 VDC) Notel
14	Reserved1	No connect
15	Reserved2	No connect
16	Reserved3	No connect
17	VPROG1	Supply 1.8-5.0 VDC
18	VPROG1	Supply 1.8-5.0 VDC
19	VPROG0	No connect
20	VPROG0	No connect

## ISP requirements for target system

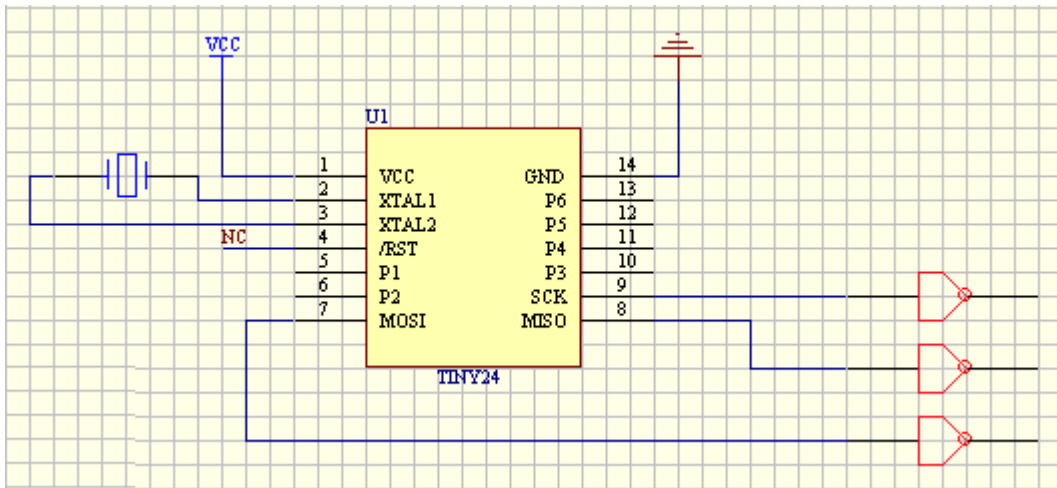
The first step to adopt ISP programming is to take into account the particularity of ISP at the design stage of target board. This point is often neglected by many people, they think it is enough as long as the control signal lines are connected to IC's corresponding pins.

### Attentions on pin connection

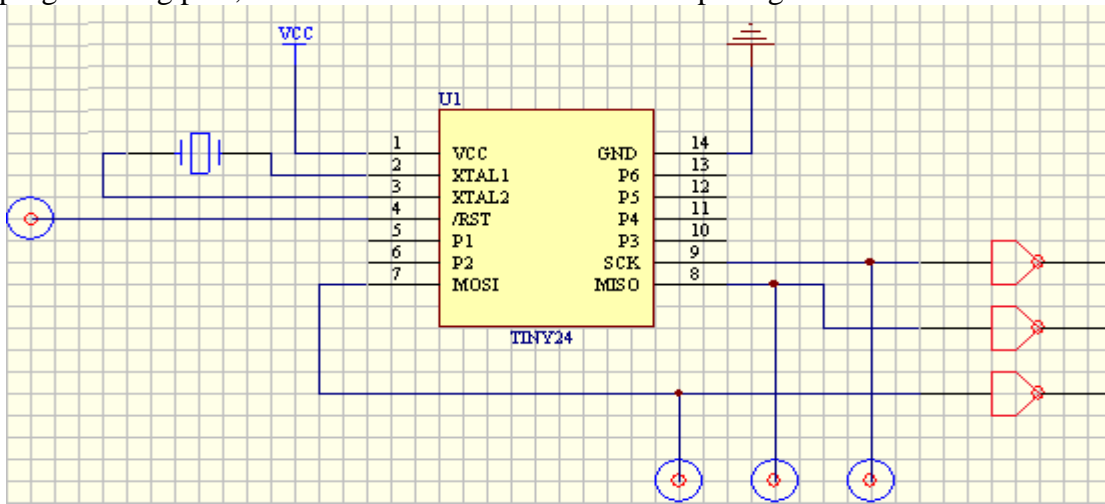
1. Pins required during the programming all should have interface terminals or measuring points in order to ensure that IS01 can be connected to them conveniently.
2. In order to prevent the required programming pins from incurring signal jump during the programming, it must be ensured that the PCB circuit board where programming pins locate is not connected with other circuits.

Below are some common circuit connection methods, we will describe their merits and demerits one by one.



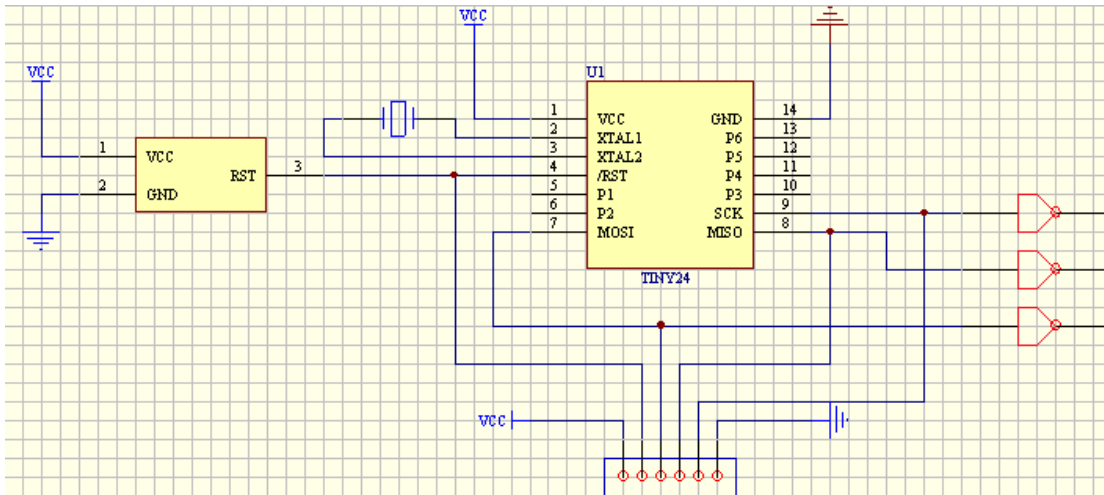


Such design in above figure makes the programmer difficult to connect with the programming pins, and also makes difficult to test the pin signals.

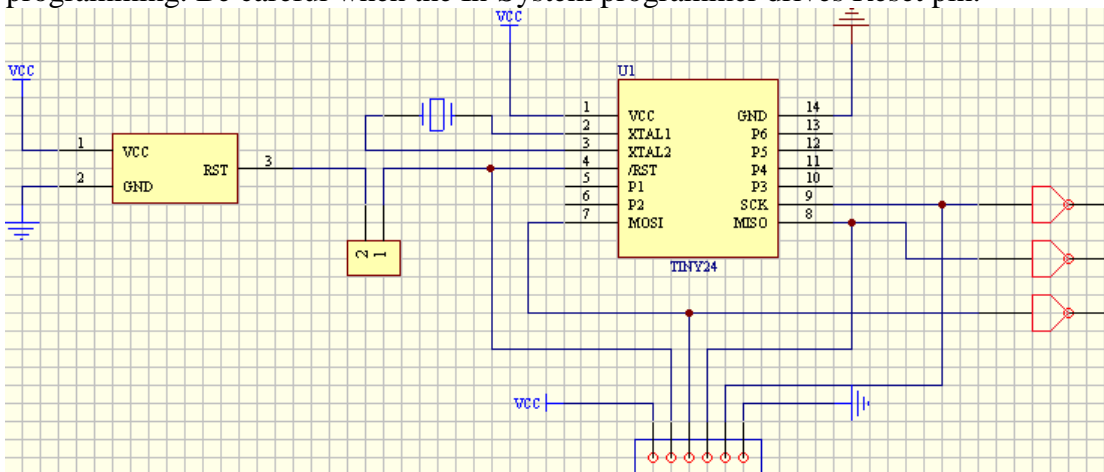


All the programming pins in above diagram have test point, therefore, it is easy to measure the signal values. But since there are no connecting terminals for programming, it is still not easy to connect with the programmer.

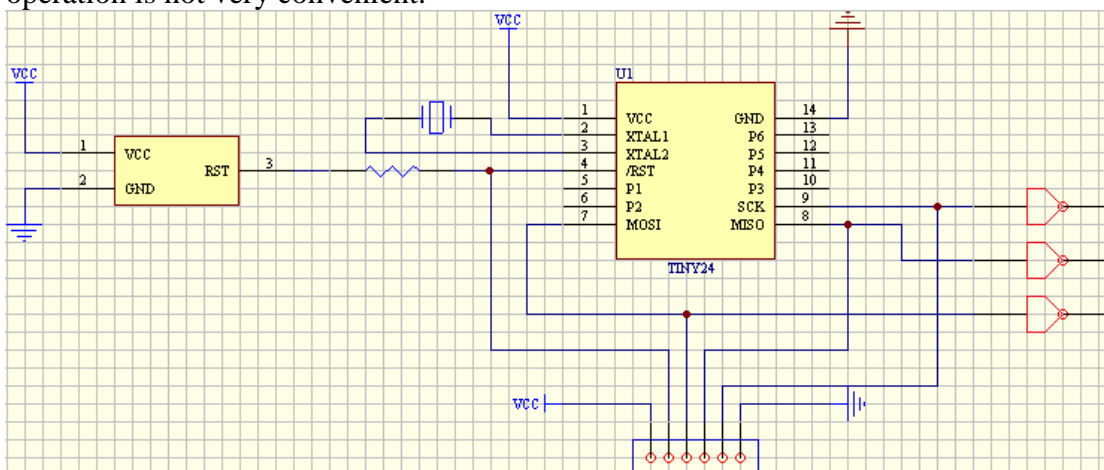




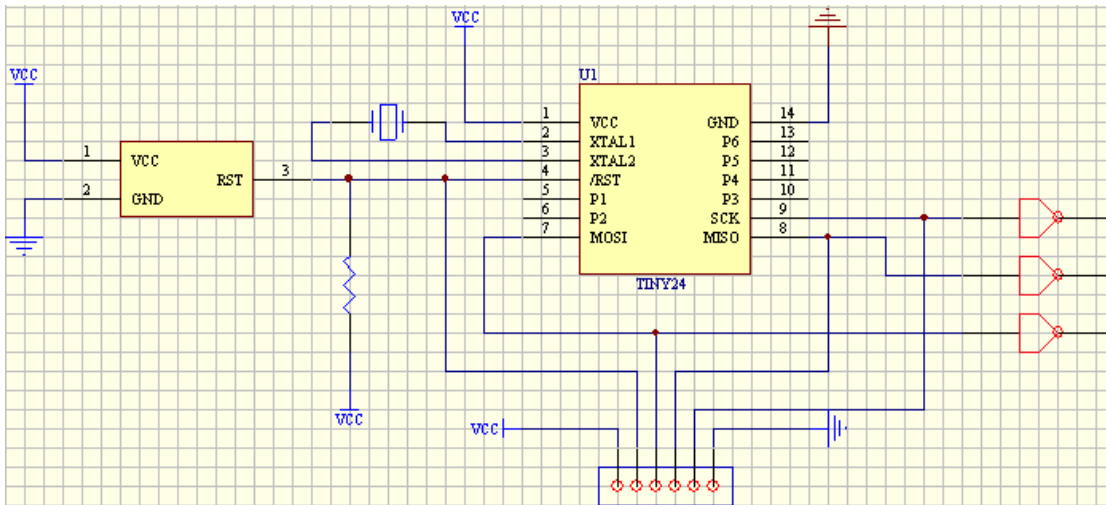
The reset generator in such connection in above diagram conflicts with the Reset pin for programming. Be careful when the In-System programmer drives Reset pin.



Such connection in above diagram eliminates pin conflicts by adding a jumper, but such operation is not very convenient.

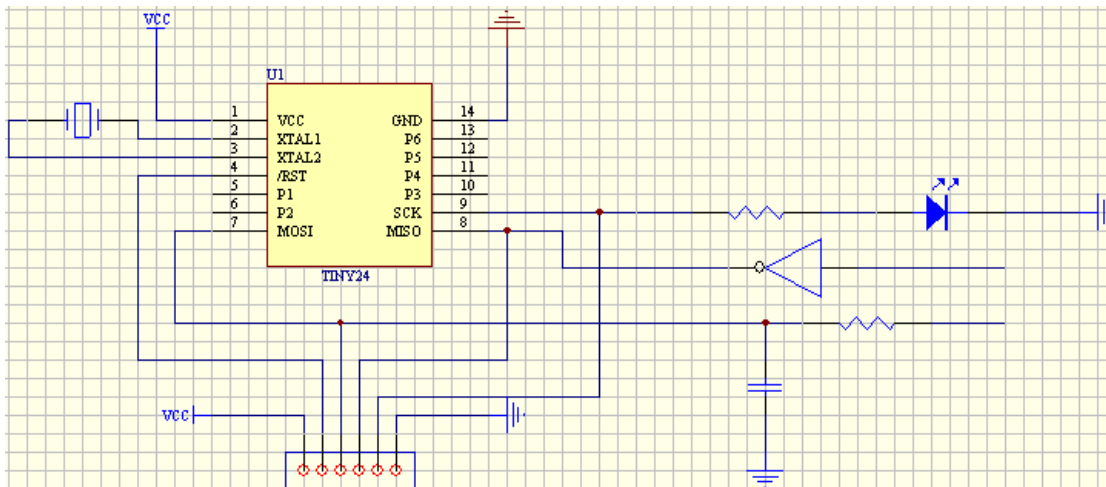


Such connection in above diagram eliminates the problem of pin conflict through adding a resistance of large value, but we still need to pay attention to the effect of pull-up resistance/pull-down resistance on input level signal of Reset pin.

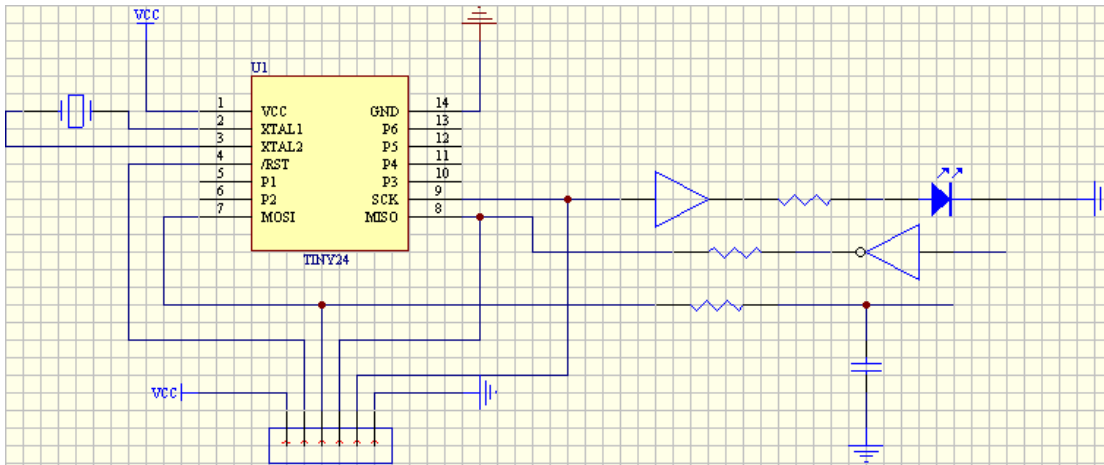


Such connection in above diagram is relatively reasonable, it settles the conflict problem through a pull-up resistance, and makes the equipment work well. It is important to note that some Resetters will automatically delay for a length of time after detecting external reset signal in order to meet the need of external equipment in this way.

## Examples of common problems



There are three problems in above diagram ① LED lamp will extract large current from ISP programmer; ② external input signal on MISO pin conflicts with ISP programming signal; ③ capacitance on MOSI pin may affect the value of level signal output from this pin.



This is the circuit diagram after improvement, it can be seen from the diagram, addition of a buffer before LED can alleviate limitation of IS01 on output current; the series resistances before MISO can eliminate signal conflict; changing connecting position of capacitance eliminates effect of filter capacitance on level signal.

### Other matters needing attention

- The watchdog circuit in the programming process can be triggered normally.
- If the bootloader of the chip need using during the programming, the bootloader must have been stored inside the chip before ISP programmer works, except that the bootloader has been stored in the mask ROM inside the chip or set by factory. For example, if there is a bootloader set by factory inside the chip, but if the space where bootloader stored by this chip is erased or modified by ISP, it is impossible to perform other ISP operation. In addition, some chips need special bootloader which makes them work well.
- If the chip contains encryption fuse operation or special function inhibit bit, it may contain online programming inhibit function, therefore, the programming last time may result in ISP function being disabled, please refer to chip manual for detail.
- If the serial working cable for ISP is too long, this may also make ISP unable to work normally. In this case, try selecting medium and low speed in the software.
- Some equipment manufacturers have indicated whether this chip has ISP function or not in the name of the chip. For example, some manufacturers use NIS to indicate the device doesn't support ISP function while use IS to indicate the device does.

### Examples for connection of several common serial communication protocols

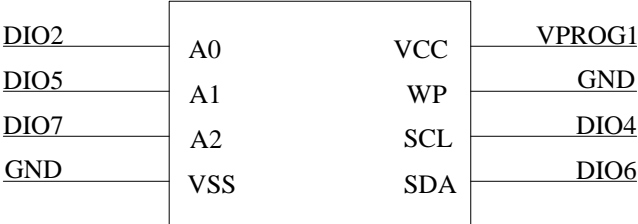
#### M25P64

M25P64 is a large-capacity serial ROM made by STM company, and programmed through SPI protocol, in this example, the chip packaging mode adopted is SO16, the schematic connection diagram is shown as below:



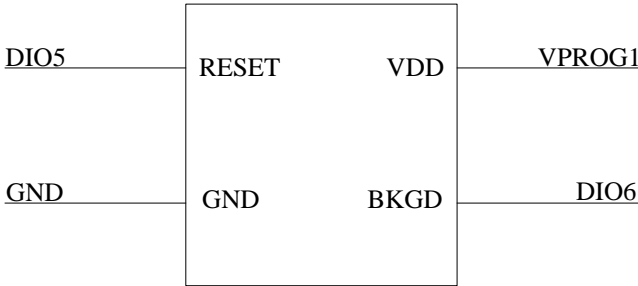
### 24C04

ST 24C04 is a universal serial ROM made by STM company, and programmed through I2C protocol, in this example, the chip packaging mode adopted is SO8, the schematic connection diagram is shown as below:



### S9S08AW16

S9S08AW16 is HCS08 series low power consumption and high-performance 8-bit controller mainly widely used for consumer electronics manufactured by Freescale Comapany. This chip is programmed through BDM serial protocol, in this example, the chip packaging mode adopted is LQFP44, the schematic connection diagram is shown as below:



### Use ATE interface to control the programmer

ATE control port is used for communication with external host or integration into the automatic programming and test equipment. Therefore, besides the keyboard, the user can also operate ISP programmer through ATE interface. ATE equipment interface adopts optocoupling device to isolate signals, in order to ensure security and accuracy of signals between the two equipments. In addition, while using ATE interface, it is required to set the power voltage of external equipment signal and appropriate reference voltage point correctly. For the signals of ATE port, please refer to the diagram below:

Pin #	Signal Name	Description
1	SEL0	Optoisolated, flow select input BIT0 (5-24 VDC, 20mA max)
2	SEL1	Optoisolated, flow select input BIT1 (5-24 VDC, 20mA max)
3	SEL2	Optoisolated, flow select input BIT2 (5-24 VDC, 20mA max)
4	SEL3	Optoisolated, flow select input BIT3 (5-24 VDC, 20mA max)
5	SEL4	Optoisolated, flow select input BIT4 (5-24 VDC, 20mA max)
6	SEL5	Optoisolated, flow select input BIT5 (5-24 VDC, 20mA max)
7	OPTO_GND1	Optoisolated input ground
8	OPTO_GND1	Optoisolated input ground
9	START	Optoisolated, flow START, (5-24 VDC, 20mA max)
10	RESET	Optoisolated, flow RESET, (5-24 VDC, 20mA max)
11	PASS	Optoisolated, flow PASS, (open-drain, 25mA max)
12	FAIL	Optoisolated, flow FAIL, (open-drain, 25mA max)
13	OPTO_GND2	Optoisolated output ground
14	OPTO_GND2	Optoisolated output ground
15	OPTO_VCC	Optoisolated output VCC(24V differential max)
16	OPTO_VCC	Optoisolated output VCC(24V differential max)
17	VCC	Non-isolated isp system power supply
18	VCC	Non-isolated isp system power supply
19	GND	Non-isolated isp system GND
20	GND	Non-isolated isp system GND

## Time sequence of ATE interface

For the specific operation methods and communication protocols of ATE port, please contact XELTEK.

## Installation of software

If you use Xeltek online programmer based on USB port for the first time, the content of this chapter will help you to correctly install the application software for the programmer and connect programmer hardware. USB equipment is a kind of plug & play equipment, in the first time installation, Windows will invoke “add new equipment wizard” and scan all available INF files, attempting to find appropriate driver.

### Download the installation software from the Internet

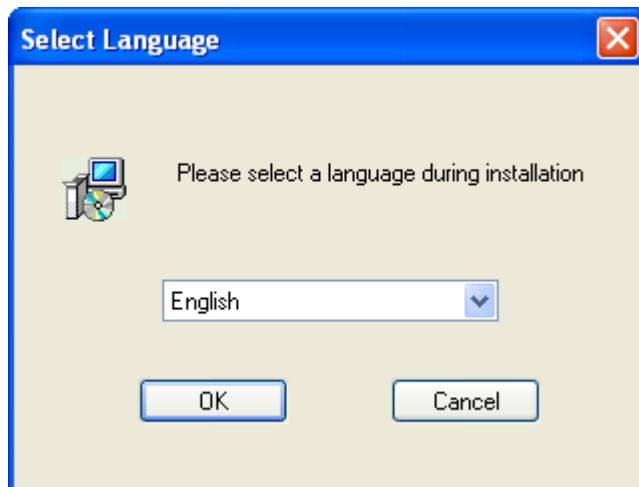
You can download the specific software for a certain model at Xeltek website: <http://www.xeltek.com>. It is usually a self-extracting file. You only need to run this file to setup the software.



## Setup Process

The setup procedure goes step by step and you can change the default settings according to your needs in the setup process.

**Step 1:** Select the language that you want to use in the setup process, as shown in the following language selection interface.

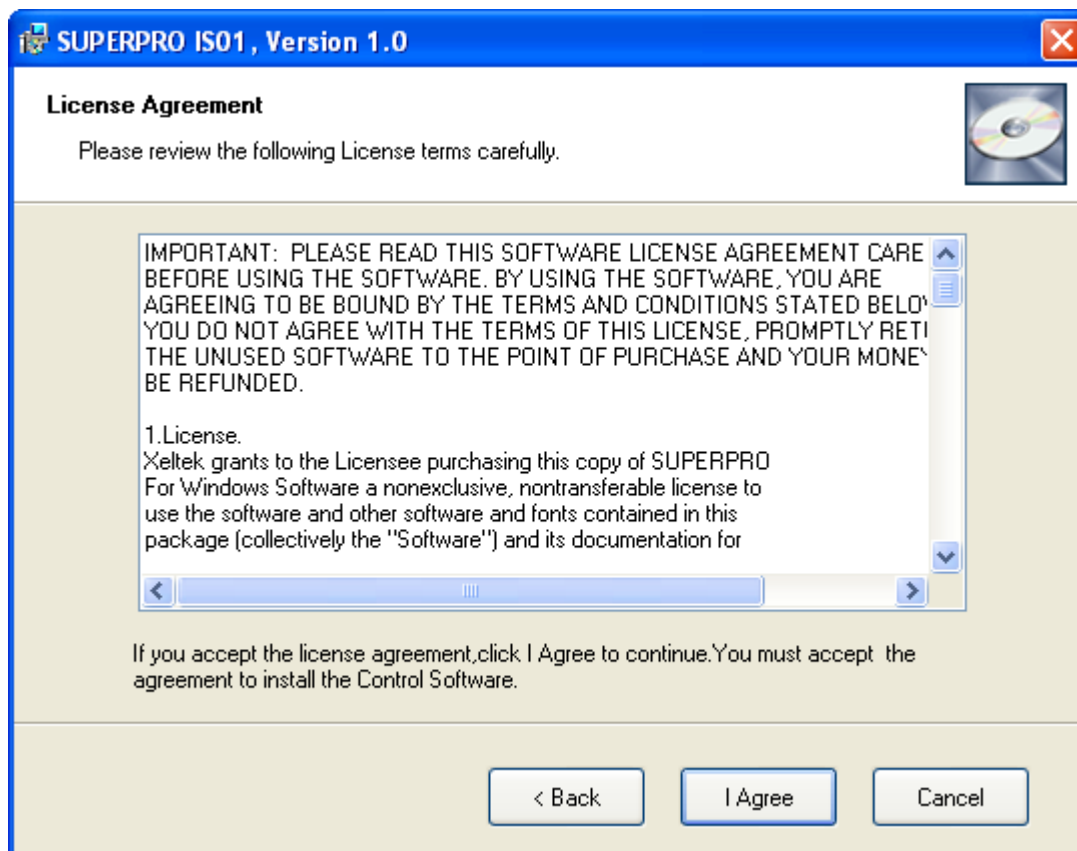


Click the "OK" button and a dialog box of "Preparing to install" appears.

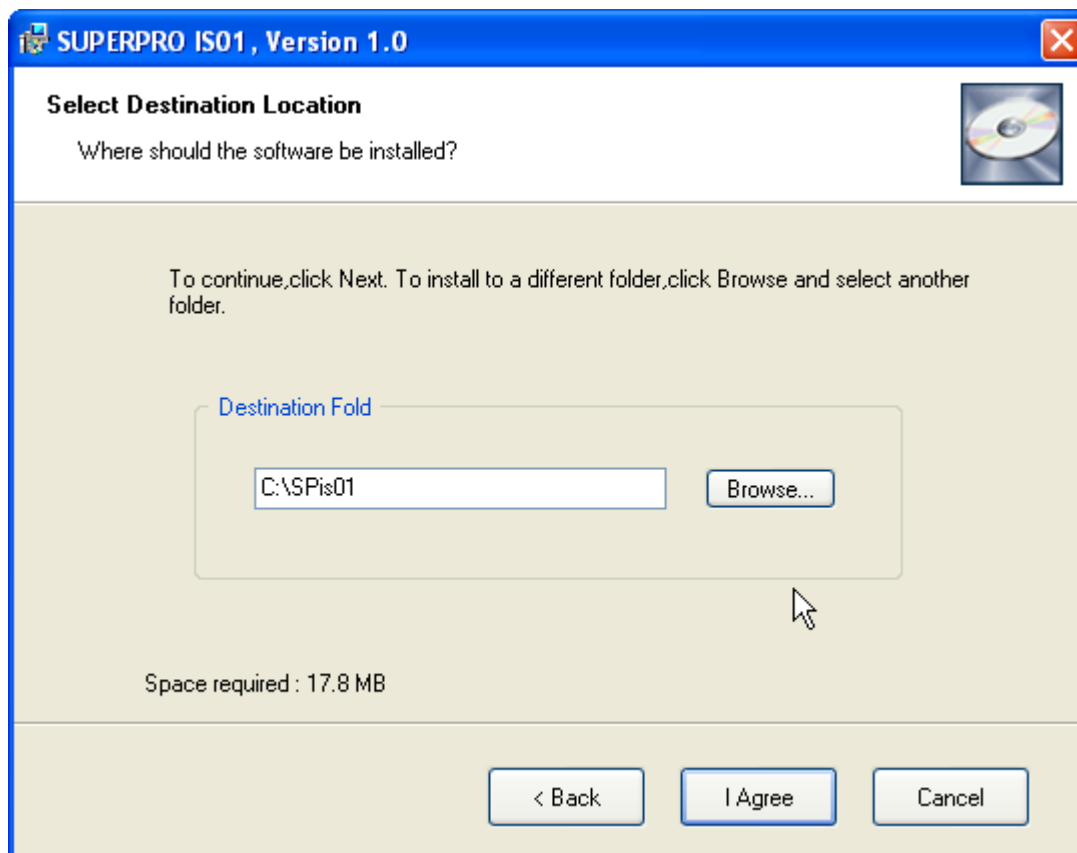




**Step 2:** Click the "Next" button. The License Agreement appears.

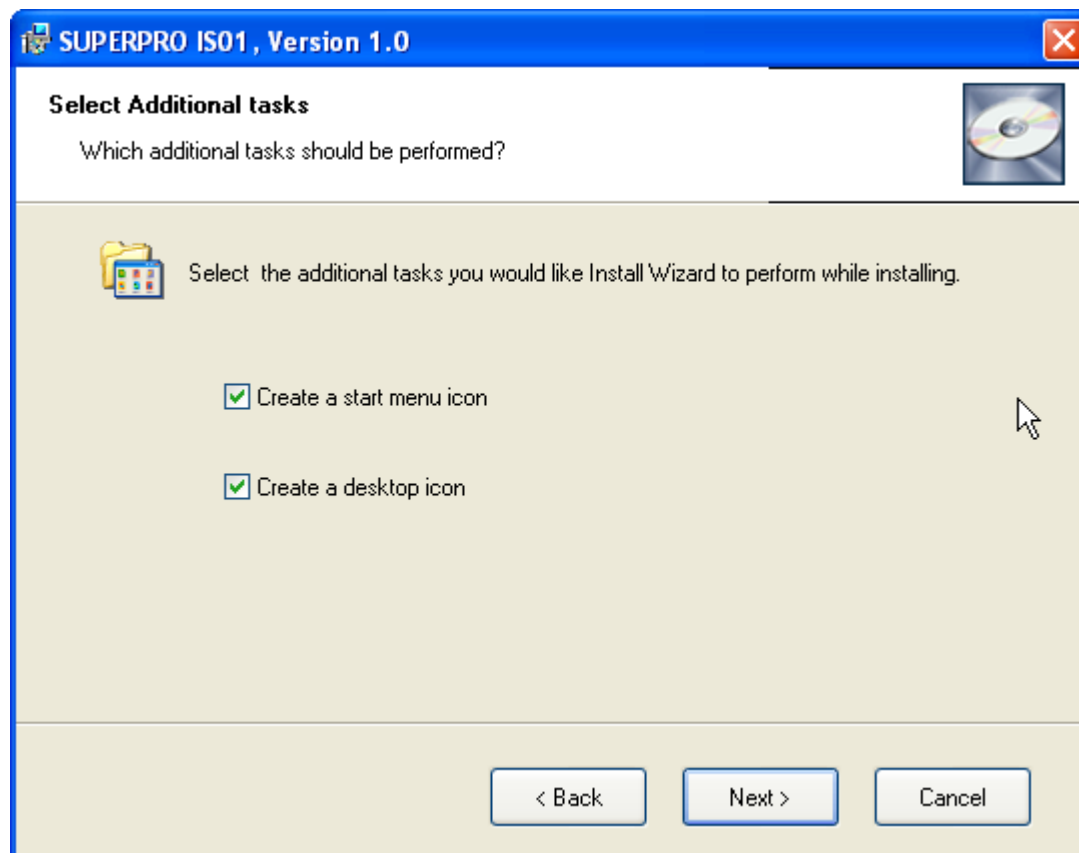


**Step 3:** Please read the license agreement carefully and click “I Agree” to continue the installation. Otherwise, click Cancel to exit the installation.



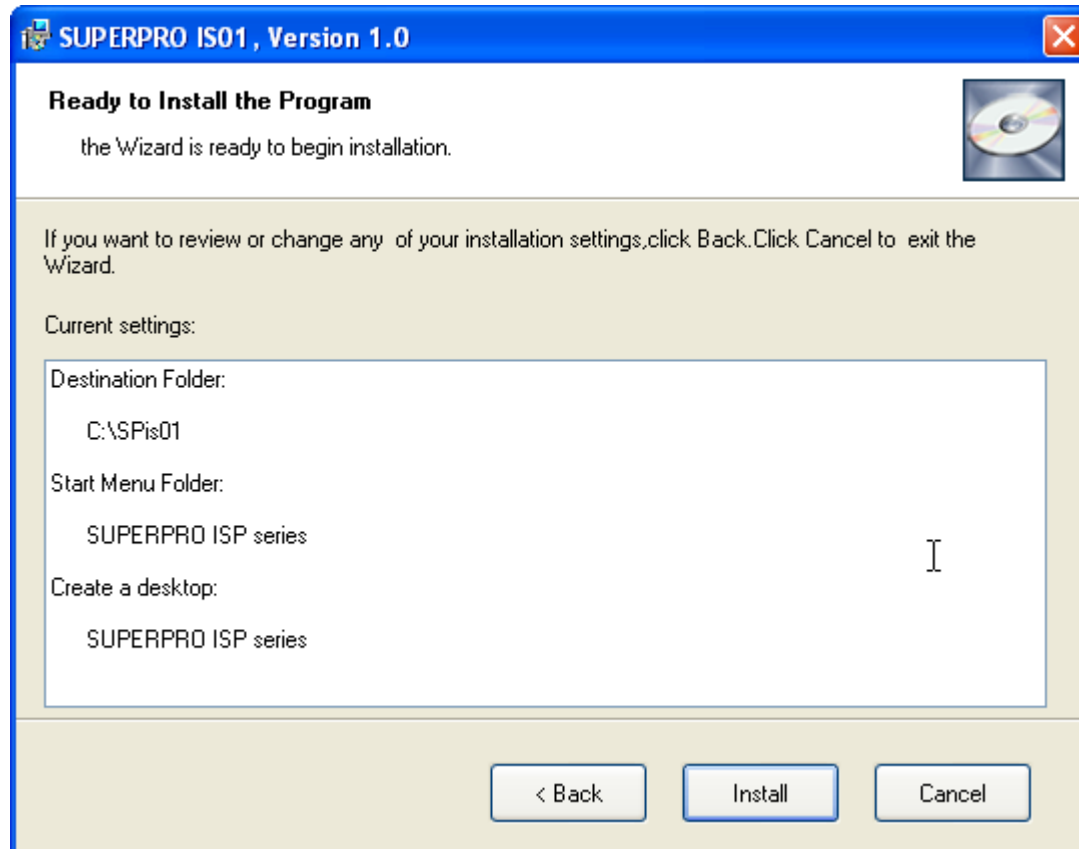
**Step 4:** You can select the installation path in this step. Click the “Browse” button to change the default destination folder. Click the “Next >” button to continue the installation.

### Step 5:



You are asked whether to create a start menu icon and a desktop icon in this step. If you select the default setting, click the “Next” button to continue.

### Step 6:

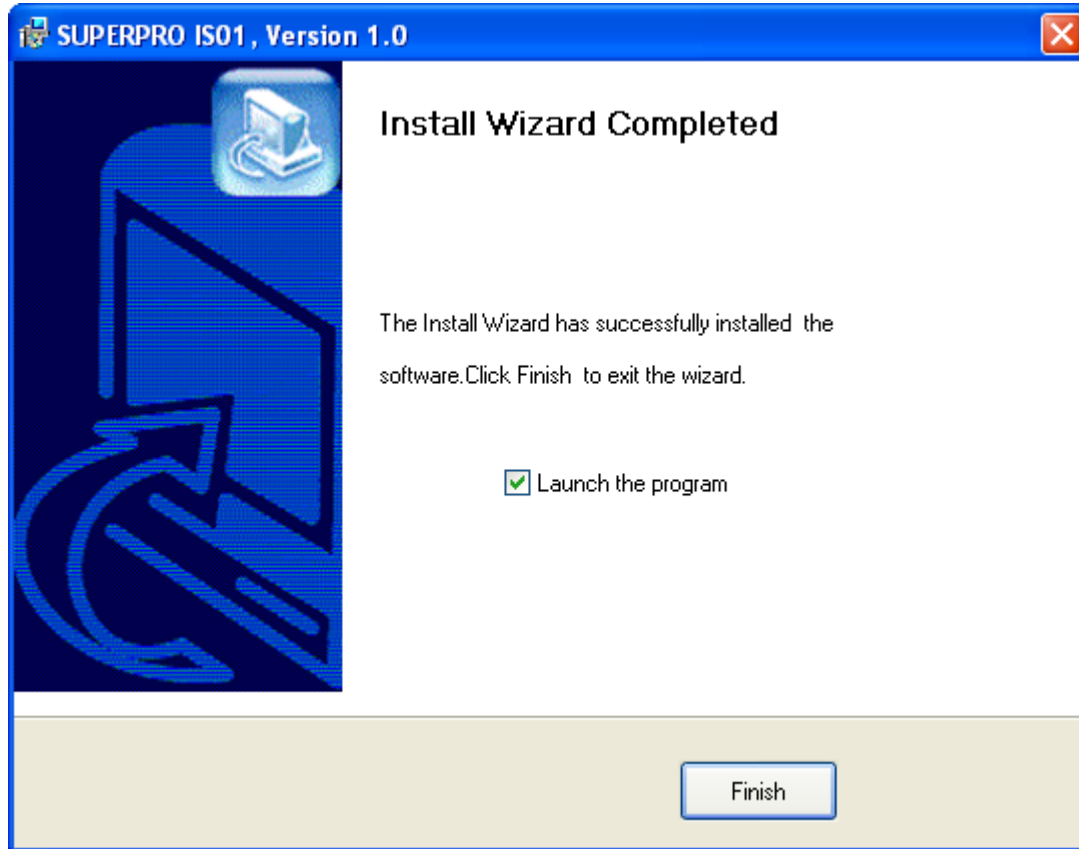


Check the above settings that you set in the setup process. Click the “Install” button to proceed to the next step.

### Step 7:

In this step, the files are being copied, the drivers are installed, the software is registered, etc.

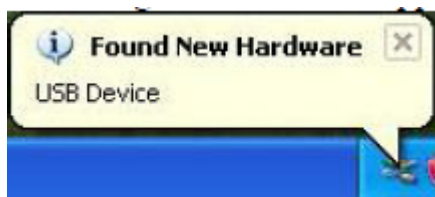
### Step 8:



Click the “Finish” button to finish the setup.

### The hardware and driver installation

Connect the programmer module to your computer through the USB port. Turn on the programmer power switch. After the application software is installed, the driver installation for the first time connection between the computer and the programmer module becomes very simple. You only need to wait until new hardware wizard finishes the installation. The following interfaces will appear on the screen one after another:





The actual interfaces may be somewhat different from the above ones. Anyway the user should wait until the installation finishes. Please don't run any other programs during the installation.

### **Running the application software for the programmer**

The application software for the programmer has a standard Windows UI, including pull down menus and buttons, etc. The user should have some knowledge about the MS Windows OS and be able to use the computer mouse well. After the application software for the programmer is started, it will communicate with the programmer immediately and start the initialization. If the communication fails, please check according to the following steps:

- One programmer module. Check the connection between the programmer hardware and the computer, and turn on the power switch.
- Follow the correct installation steps. If the programmer hardware is connected to the computer before the programmer application software is installed, the computer will detect the new hardware and the following window will appear.



Please click the “Cancel” button lest the OS should install the driver with a compatible ID and cause some problems.



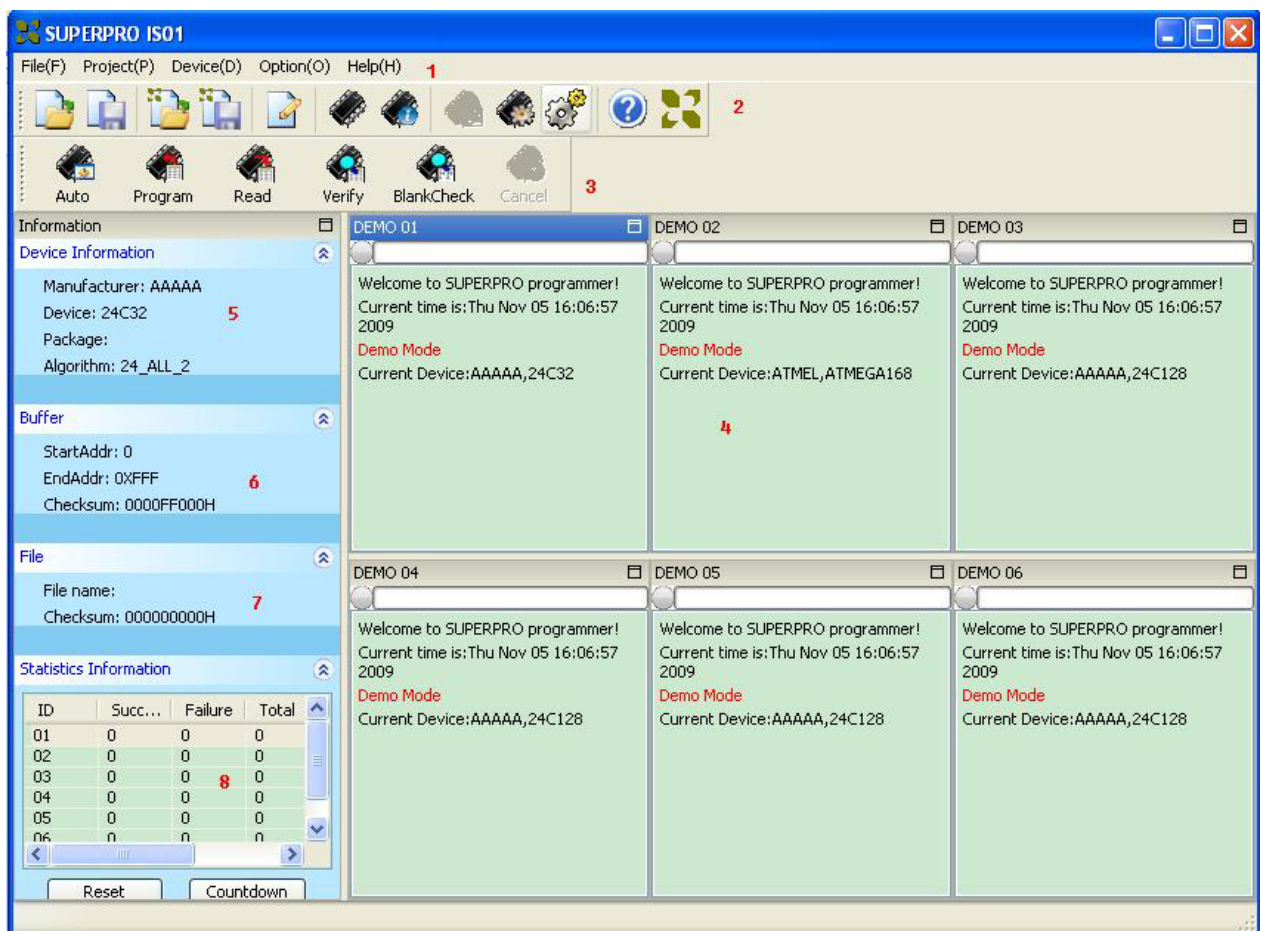
## Quick start

This chapter will help you to understand the whole process of chip burn-recording operation from beginning to the end, the content includes:

- User interface of application software
- Steps of burn-recording device

Before you are ready to operate the burn-recording device of the programmer, please make sure that you have properly installed the programmer, and the computer has successfully communicated with the programmer.

## User interface of application software



- |                         |                           |                           |
|-------------------------|---------------------------|---------------------------|
| 1. Menu bar             | 2. Main toolbar           | 3. Operation toolbar      |
| 4. Log window           | 5. Device information bar | 6. Buffer information bar |
| 7. File information bar | 8. Statistic window       |                           |

Note 1: The number of subwindows in the log window is determined by the number of modules set by the user. If there is no special description, the following description is

made all with respect to current window operations (i.e. highlighted window, ATMEL, AT24C64A here)


Note 2: It is suggested to start programmers before software when you select multi-module mode.

## Steps for chip burning

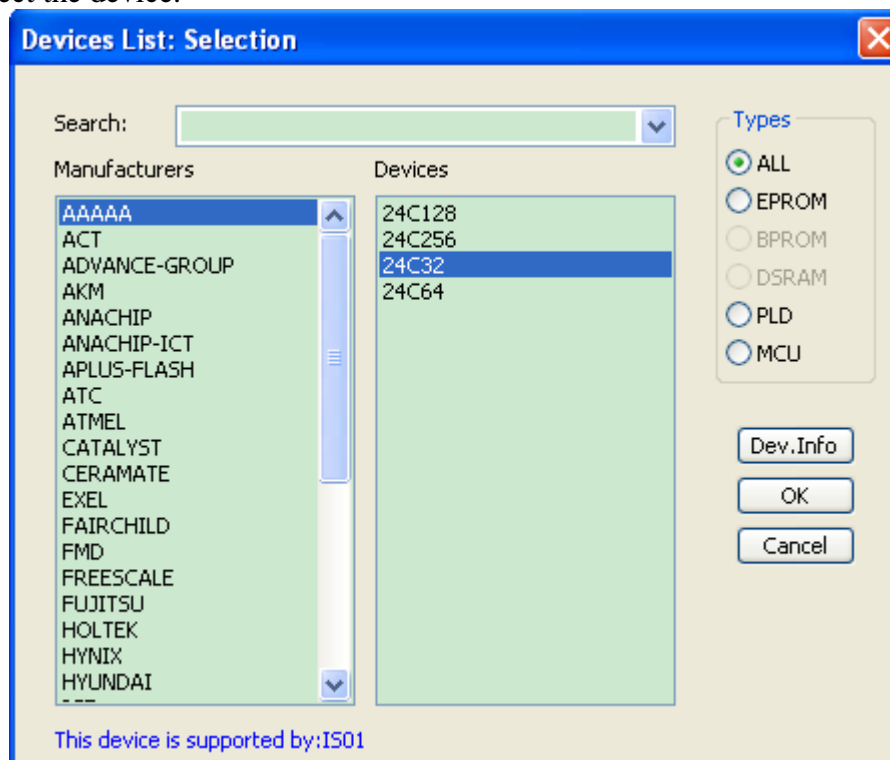
### Hardware preparation

Before you are ready to burn devices by the programmer, please make sure that you have properly installed the programmer, and the computer has successfully communicated with the programmer. (after communicating successfully, the log window will display “SUPERPRO IS01 start”, otherwise, the program will enter demo mode).

### Select device

Click “select device” on the device menu bar or  on the tool bar, then a dialogue box for selecting device pops up.

First select device type, for example, E/EPROM, BPROM, DSRAM, PLD or MCU, then select manufacturer and device name, then click OK button. You can also key in device name in the Search edit box to narrow down the select range in order to quickly select the device.



Note: IS01 does not support BPROM, DSRAM.

## Load data into buffer

Programming device is a process to burn the data in the buffer into the storage unit of the chip according to the manufacturer's requirements. There are two ways for loading of data into the buffer.

### 1) Read from file

Select "Load" from "File" on the main menu, select appropriate file, format, and type in "Load File" dialogue box, after confirmation, the data file is loaded. The user can check the data in the buffer edit window.

### 2) Read data from master slice

After selecting the device, place the master slice, click "Read" function item on the operation tool bar, then the data in the chip will be copied into the buffer. At this time, the user can enter the buffer edit window to check if the data is correct or not. This data can be stored in a disc for the need later.

Attention: some devices do not have read-out function, or have been encrypted so that data can not be read out from master slice.

## Set options

### 1) Operation option, including:

- ID Check: Checks ID of device before chip burning.
- Automatic Increment: when the user selects chip burning, he writes the serial number in the form of accumulative number at the designated position so that each chip after burning has different mark number.
- Change the starting and ending address of the area of Device that needs to be burn-recorded
- Verify Mode selection. Select specific voltage value of VCC for verification in order to verify correctness of chip burn-recording.

2) Edit Auto mode. In the operation tool bar, all devices have a basic batch process operation "Auto", its function is to execute other operation functions of devices in the already edited sequence. Generally, Edit Auto mode is selected for all devices.

Erase

Blank check

Program

Verify

Security or Protect

3) Dev. Config. For the chip with configuration words, the configuration words must be set before burning the device to ensure the burned chip can work in the user's target system. The configuration words of some devices are contained in the user's data file and are automatically filled in the buffer of configuration characters upon import of data file.

4) Dev. Info. After selecting the device, the dialogue box for device information pops up to display the important information of relevant device. Some devices have special requirements for burning, or the algorithm for burning has special convention, so the user should adjust the operation steps or buffer data after carefully reading the requirements or conventions. When adapter is needed to support burning, corresponding adapter information is displayed (IS01 shows how to connect IS01 programmer and some instructions) .

5) Production mode. The programmer checks the socket status all the time, once the chip is

placed completely, it automatically starts AUTO command without manual key push to start operation in order to facilitate the user to batch burn the chips (IS01 production mode can only be used in offline mode).

## **Programming**

The user can buy the adapter according to the prompting information and place it into the chip correctly.

The operation steps for the burning device are as follows:


- Blank Check. If the chip is new, this step can be omitted
- Program. Burn the data in the buffer into the chip.
- Verify. This step is necessary, only when the verification is successful can the chip burning be considered as correct. Some chips do not provide unit-to-unit verification function, but just provide accumulative verification function similar to VerifyCRC. And very few chips even do not provide accumulative verification function.
- For the electrically erasable chip already with content, Blank Check will fail, in this case be sure first to perform Erase operation. Some EEPROM devices can directly overwrite without the need of erasure in advance.
- If the device needs to be protected, add Security or Protect function after verification. Notice that the protection of some devices requires to set configuration characters for the device in advance, please refer to the detailed description of functions.

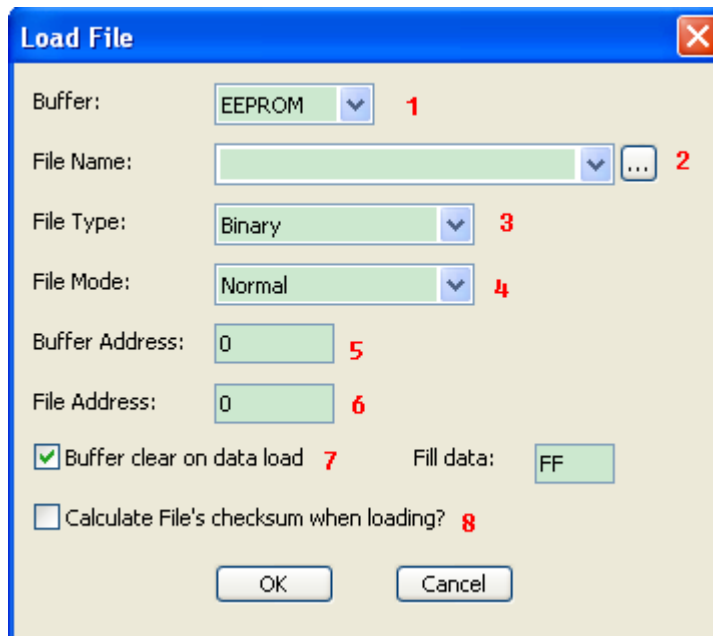
The user can select Auto function to do all operations once for all.

## **Detailed description of functions**

### **Browse menu**

#### **Load file**


Select “load file” on the menu bar, or click  on the tool bar, then the dialogue box for file loading pops up.

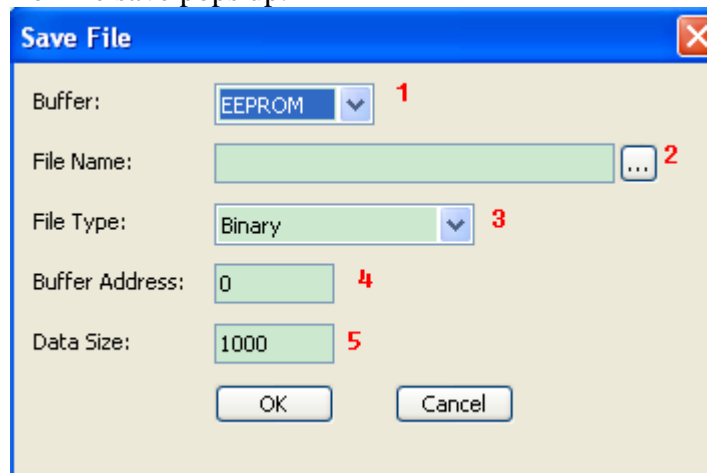


1. Data buffer: the data buffer of the device is divided into one or more areas. If there are several data buffers displayed after selecting the device, it is required to explain the meaning of the data buffers according to the names of data buffers and data manual.
2. File name: designate the file with loaded data. The user can select file by file dialogue box which pops up when button “...” is clicked or from file history record which can be browsed when drop-down arrow is clicked.  
Note: After selecting the file, it is still needed to select the appropriate type of data file, then it can be ensured that the data is loaded correctly.
3. Type of file: include Binary, Intel Hex (extension) etc.
4. Loading mode:
  - <Normal>: the whole file is loaded.
  - <Even>: First byte of every two bytes is taken and the second byte is abandoned.
  - <Odd>: Second byte of every two bytes is taken and the first byte is abandoned.
  - <1st byte of 4>: First byte of every four bytes is taken and the rest three bytes are abandoned.
  - <2nd byte of 4>: Second byte of every four bytes is taken and the rest three bytes are abandoned.
  - <3rd byte of 4>: Third byte of every four bytes is taken and the rest three bytes are abandoned.
  - <4th byte of 4>: The fourth byte of every four bytes is taken and the rest three bytes are abandoned.
  - <1st 2\_byte of 4>: The first two bytes of every four bytes are taken and the rest two bytes are abandoned.
  - <2nd 2\_byte of 4>: The last two bytes of every four bytes are taken and the rest two bytes are abandoned.

5. Buffer address: the start address where data is loaded into in the buffer.
6. Offset address of file: some types of file have non-zero start address (data offset address), thus it may be required to input the file start address when loading the file. Please input correct address data in the Edit box. Incorrect file offset address will result in filling in the buffer head by FF (or incorrect data), while incorrect big offset address will result in data flooding or system failure.
7. Check if the buffer is to be cleared when loading the file.  
If the user selects “Yes”, then he must indicate which value is used to erase the buffer.
8. Check if the check sum is calculated when loading the file.

## Save file

Select “save file” on the menu bar, or click  on the tool bar, then the dialogue box for file save pops up.




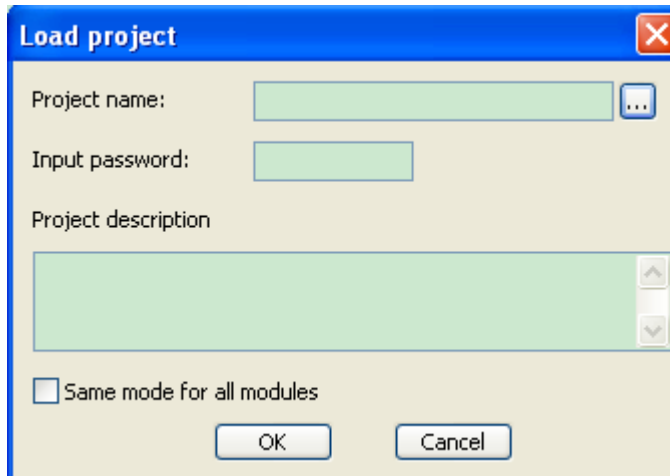
1. Data buffer: select the buffer data to be saved
2. File name: select to save the buffer data into what kind of file.
3. Type of file: include Binary, Intel Hex (extension), Motorola S record, Tektronix Hex, etc.
4. Buffer address: starts to save the data from the start address specified in the buffer.
5. Data size: specify the size of data to be saved.

## Exit

Close the programmer software and return to the operating system.

## Load project


Select “load project” on the “Project” menu bar, or click  on the tool bar, then the dialogue box for project loading pops up.

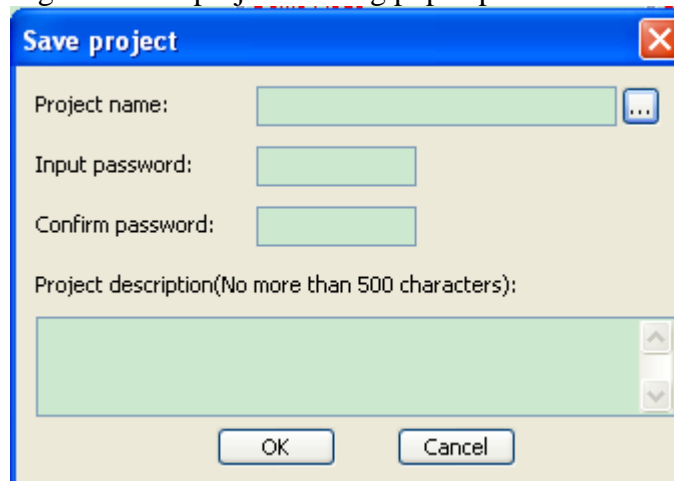


After selecting the project and inputting correct password, the project is loaded.

If the “all modules adopt the same mode” is chosen, then this project will be loaded into all the modules.

## Save project

Select “save project” on the “Project” menu bar, or click  on the tool bar, then the dialogue box for project loading pops up.



Project name: specify the name of project to be saved.

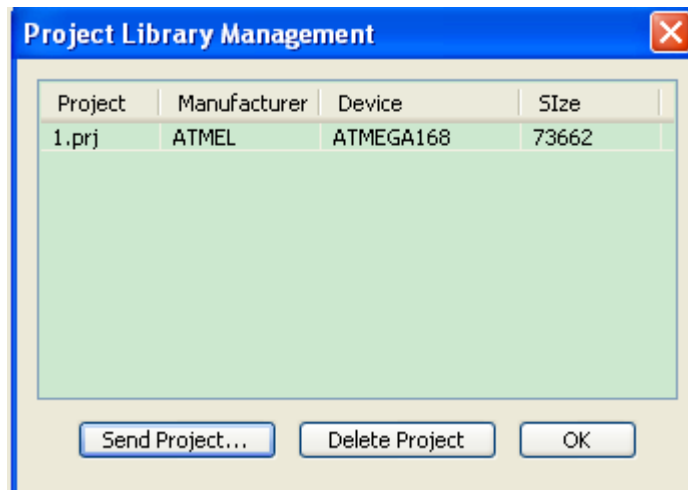
Password: for data security, the user can add password in the project file.

File description: you can add simple description information for your project (not exceeding 1000 bytes).

Note: the suffix name of project files must be .prj.

## Project library management

Select “Project library management” on the menu bar. Then the dialogue box for project library management pops up.



The project information in SD card of the programmer is displayed in the project list (project name, manufacturer name, device name and size)

Send project: Send the local project file down into SD card (note: before sending the project, you must ensure that SD card has been inserted)

Delete project: delete the specified project file in SD card.

### **SD card operation**

#### **Set password protection**

After setting password, SD card will be protected. SD card protected by password can only be accessed after unlocking (the password must be six digits from 0~9).

#### **Cancel password protection**

After canceling password protection, the project in SD card will be cleared. SD card protection is cancelled.


#### **Unlock**

Input correct password and SD card is unlocked. And you can normally manage SD card project library in management project library. But this is only limited to this time operation, the card still needs to be unlocked in next time operation.

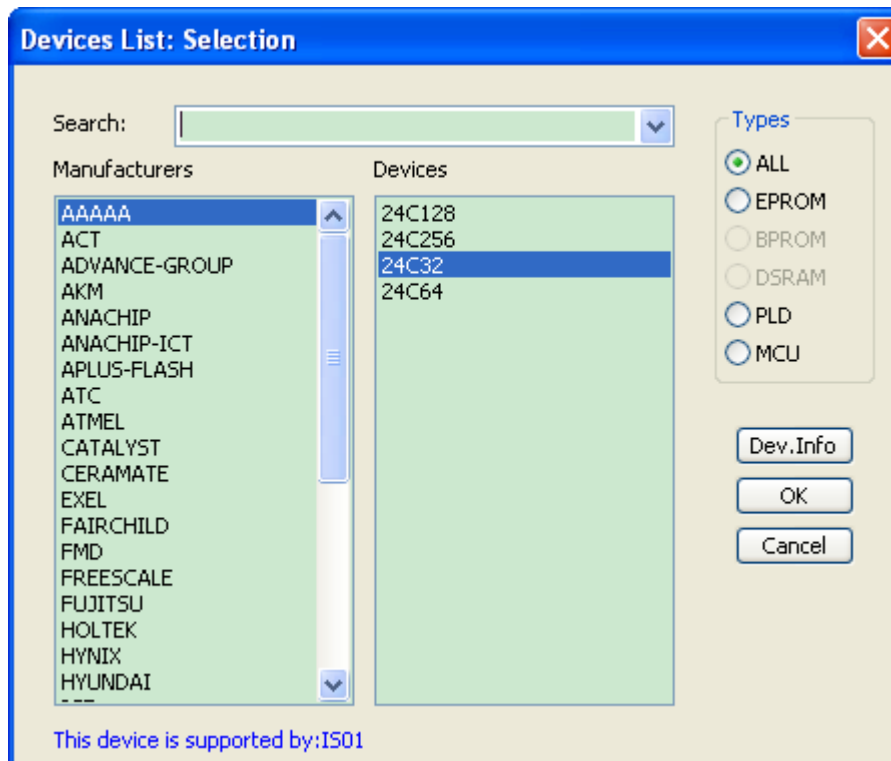
#### **Erase SD card**

If you forget the password when you set it previously, you can select to erase SD card. (Note: SD card cannot be reused if it is not treated specially after it has been erased)

#### **Select chip**

Select “select chip” on the menu bar, or click  on the tool bar, then the dialogue box for chip selection pops up.





The dialogue box for “Device” includes such controls as manufacturer browse box, device name browse box, type select button and OK and Cancel buttons as well as Search Edit box etc. How to select a device?

- Select device type among the type select buttons, there are five major types: E/EPROM (EPROM, EEPROM, FLASH etc.), PLD, B/PROM, DRAM/SRAM, MCU. If you are not sure, you can select “ALL”.
- Assign manufacturer name in the manufacturer browse box.
- Assign device name in Device Name browse box and just press “OK” button.

Since the name printed on the chip may include such parameters as speed, temperature, package etc., when you select a name that is not totally identical with the name in the dialogue box of “Device”, you should neglect these parameters or ask for help.

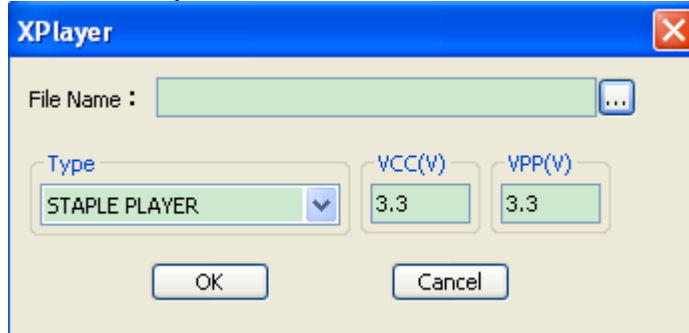
Use of Search edit box: it may be very inconvenient to directly select the device due to large number of devices and difference in names of chip, in this case we can use Search edit box to help selecting. As long as you input several key characters, then the number of devices can be greatly narrowed, thus facilitating fast search of the device. For example, when you want to find 89 series single-chip microcomputer, input characters “89” in the Search edit box, then all the devices containing characters “89” will be listed. The software is sensitive to the sequence of characters “89” when making a search, but it will neglect presence of other characters in the box. For example, character string “819” and “DA8S9” will be listed since they meet the requirement, while characters “98” will be neglected since it does not meet the requirement.

**Note:** Some chips can not be used only after authorization is given, after selecting this chip, the log window will remind “this chip needs authorization!”

## XPlayer

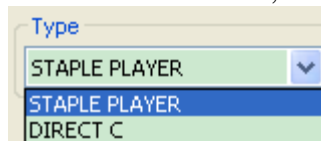
XPlayer is a kind of special programming tool mainly for CPLD series chips produced by ACTEL, ALTERA etc., as long as the customer provides corresponding STP or JAM file (such files can be generated using such design tools as ISE, Quartus II etc.), the corresponding device can be programmed.

Select “XPlayer” on the “Device” menu bar, then the dialogue box pops up.



File name: select the needed STP or JAM file.

Type: 1.STAPLE PLAYER; 2. DIRECT C




The universality of STAPLE PLAYER is comparatively better than DIRECT C, while its speed is lower than the latter.

VCC: working voltage of chip

VPP: programming voltage provided by the programmer

## Buffer

Select “select chip” on the menu bar of “Device”, or click  on the tool bar, then the dialogue box for buffer edit pops up.

HEX/ASCII data buffer

The data width of each unit is 8 bits or 16 bits according to the chip.

**Edit Buffer**

☒ Buffer Clear at IC Change 
 ☒ Buffer Clear at Data Load 
 ☐ Buffer Save when Exit

OK Cancel

---

EEPROM

Address: 00000000H Range: 00000000H-000000FFH Check Sum: 0000FF00H Calculate

ADDRESS	HEX	ASCII
00000000	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
00000010	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
00000020	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
00000030	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
00000040	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
00000050	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
00000060	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
00000070	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
00000080	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
00000090	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000A0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000B0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000C0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000D0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000E0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000F0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....

Locate Copy Fill Search Search Next Radix Swap

☒ Buffer Clear at IC Change ☒ Buffer Clear at Data Load ☐ Buffer Save when Exit

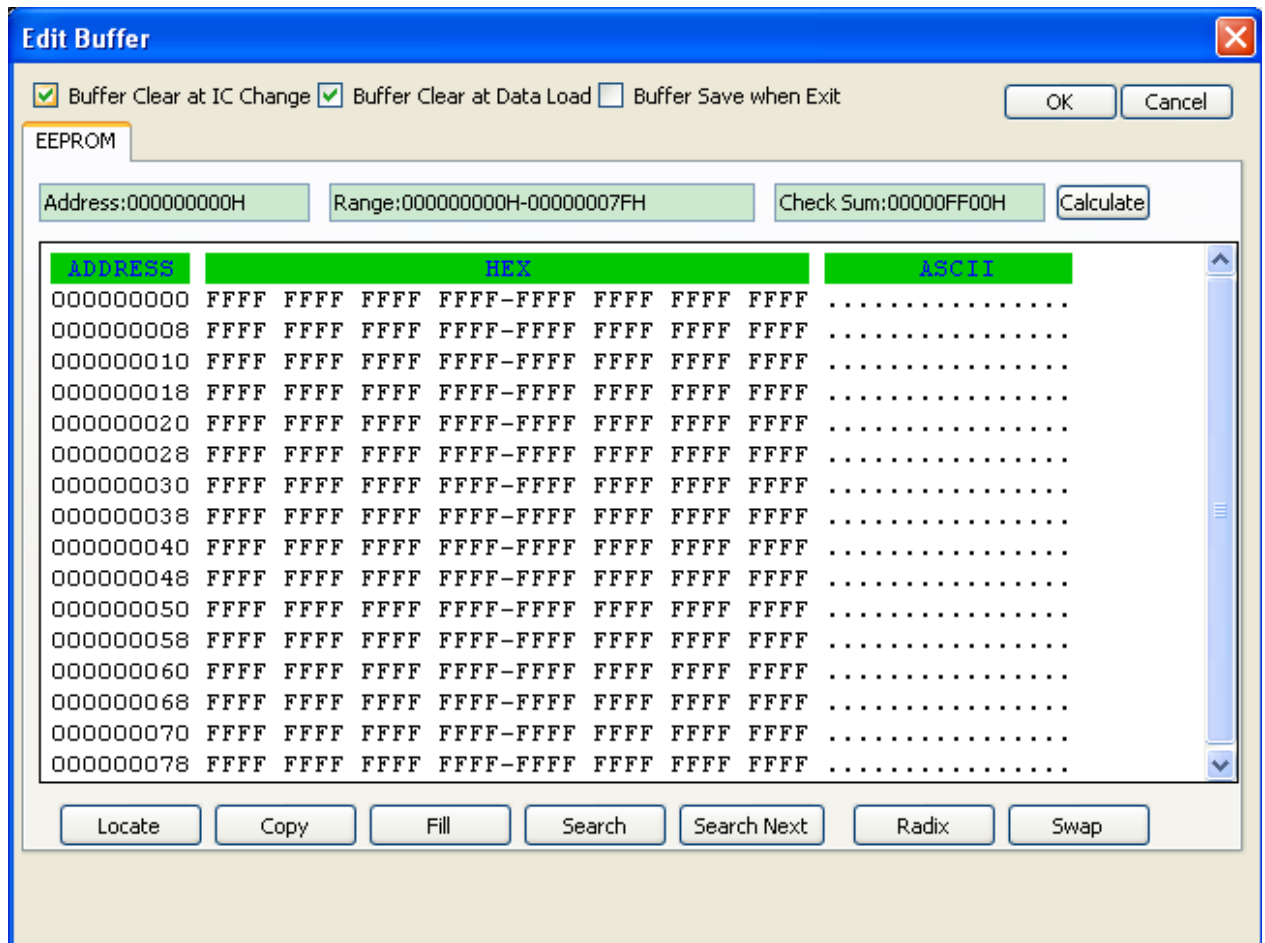
EEPROM

Range:00000000H-000000FFH

Check Sum:0000FF000H

Calculate

ADDRESS	HEX	ASCII
000000000	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000010	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000020	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000030	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000040	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000050	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000060	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000070	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000080	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
000000090	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
0000000A0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
0000000B0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
0000000C0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
0000000D0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
0000000E0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....
0000000F0	FF FF FF FF FF FF FF FF-FF FF FF FF FF FF FF	.....



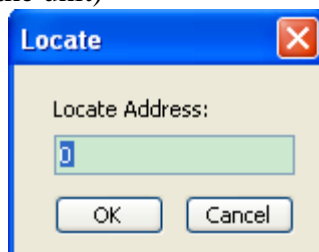
**Range:** displays the address and end address of this buffer (with character width as the unit)

**Address:** displays the position of current cursor Address:00000000H

Note: this address is the address with word width of chip as the unit

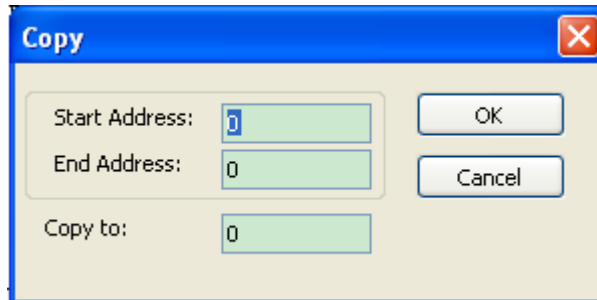
**Checksum:** see “calculation of check sum” for detail.

**Locate:** move the cursor to the designated position of the buffer (with character width as the unit)



Note: the address to be filled is hex numerical value, below is the same.

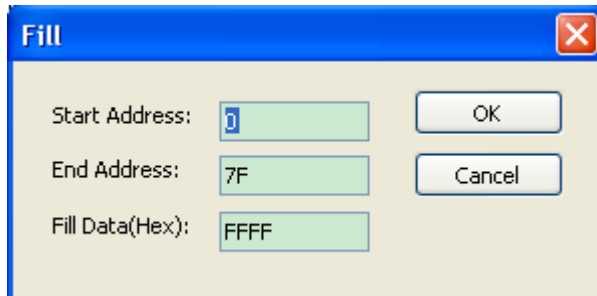
**Copy:**



A dialog box titled "Copy" with a blue header bar and a red close button. It contains three input fields: "Start Address:" with the value "0", "End Address:" with the value "0", and "Copy to:" with the value "0". To the right of these fields are two buttons: "OK" and "Cancel".

Copy the designated area of buffer to another position (with word width as the unit)

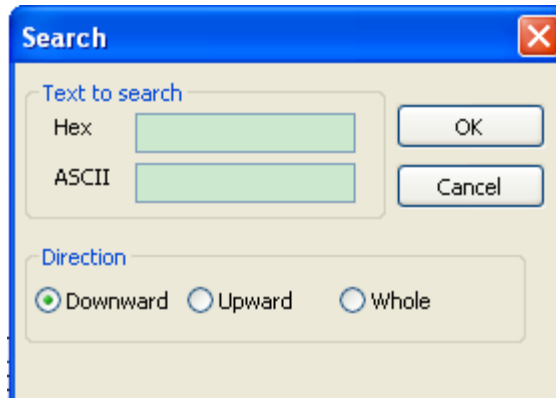
**Fill:**



A dialog box titled "Fill" with a blue header bar and a red close button. It contains three input fields: "Start Address:" with the value "0", "End Address:" with the value "7F", and "Fill Data(Hex):" with the value "FFFF". To the right of these fields are two buttons: "OK" and "Cancel".

Fill the designated area of buffer with the filling value (with word width as the unit)

**Search:**



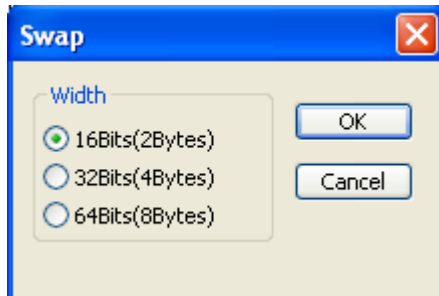
A dialog box titled "Search" with a blue header bar and a red close button. It has two main sections. The first section, "Text to search", contains two input fields: "Hex" and "ASCII". The second section, "Direction", contains three radio buttons: "Downward" (which is selected), "Upward", and "Whole". To the right of these sections are two buttons: "OK" and "Cancel".

Search the data in the buffer through hex system or ASCII, the user can select the search way (search forward from current position, search backward from current position and search backward from beginning of the buffer).

**Search next:** search next address that meets the condition according to the search way the user selects.

**Switching:** switch over the display format of address in the buffer (hex system or decimal system).

**Exchange:**



Assume the data at address 0-10(hex system) of data buffer is:

12 34 56 78 90 AA BB CC – DD EE FF 11 22 33 44 55

The character width that can be selected is:

16 bits(2 bytes), the data after exchange is as follows:

34 12 78 56 AA 90 CC BB – EE DD 11 FF 33 22 55 44

32 bits(4 bytes), the data after exchange is as follows:

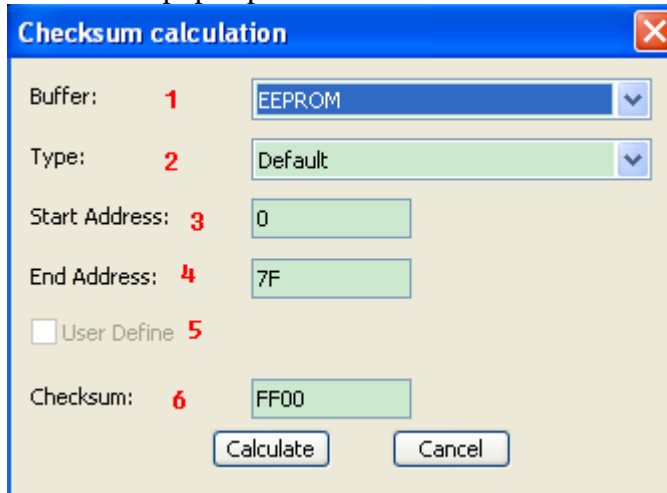
78 56 34 12 BB CC 90 AA – 11 FF EE DD 55 44 33 22

64 bits(8 bytes), the data after exchange is as follows:

CC CC AA 90 78 56 34 12 – 55 44 33 22 11 FF EE DD

### Checksum

Select “Checksum” in “device” menu bar, then the dialog box of checksum verification and calculation pops up.



When the type of data in data buffer is Data (HEX/ASCII data), the default verification and calculation method is to add up all data from beginning to the end with byte as the unit. To meet user’s special requirements, he can totalize the data in part of the addresses in a specified way.


1. Specify the buffer where the checksum is to be calculated.
2. Way for checksum calculation (sum of bytes by default)
3. The start address of checksum calculation (in bytes)
4. The end address of checksum calculation (in bytes)
5. Special method for checksum calculation, once choosing this, above selection will be disabled.
6. Checksum calculation result

## Device information

Select “device information” on the “Device” menu bar or  on the toolbar, then the current device information will be displayed.


If there is no special information, the device information only displays manufacturer name, device name, package, adapter and algorithm, this information can also be found in the left information column. If there is special information, then different content will be displayed according to different chip.

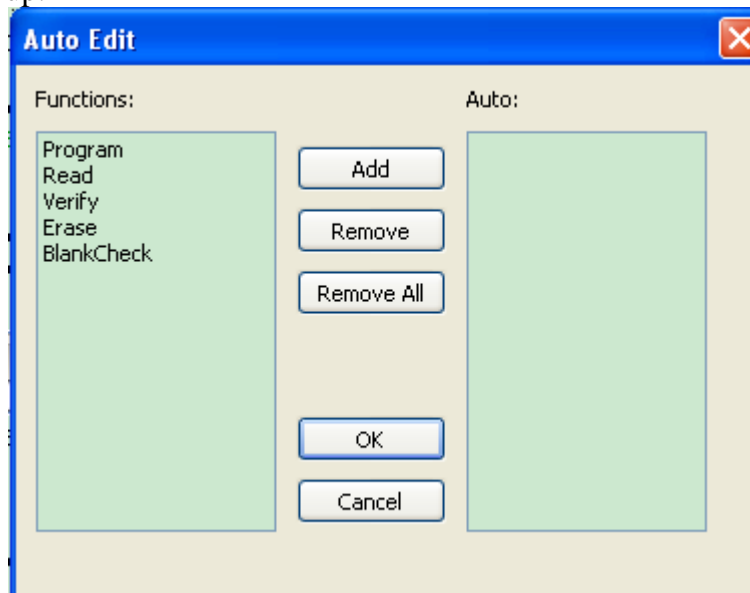
## Device Configuration Word

Select “device configuration word” on the “Device” menu bar or  on the toolbar, then the current device information will be displayed.

For the device with configuration word, the configuration word must be set before burning the device in order to ensure the burned chip can work in the user’s target system. The configuration words of some devices are contained in the user’s data file and are automatically filled in the buffer of configuration characters upon import of data file, while some devices need manual setting.

## Edit automatic processing command

Select “edit Auto” in the device menu or  on the main toolbar, the dialogue box pops up:




On the toolbar of device operation, all devices have a basic “Auto” operation, which can perform other operation functions of devices in the window of operation function items in the edited sequence, like processing batch. Open the “Edit Auto” dialogue box. All the operation function items of this device are displayed in the list box of function items on the left side of dialogue box, and operation function items and execution sequence which the operation items execute “Auto” in the “Auto” list box on the right side.

Add: Add operation function items from function list to auto list (it can also be added by double clicking this item).

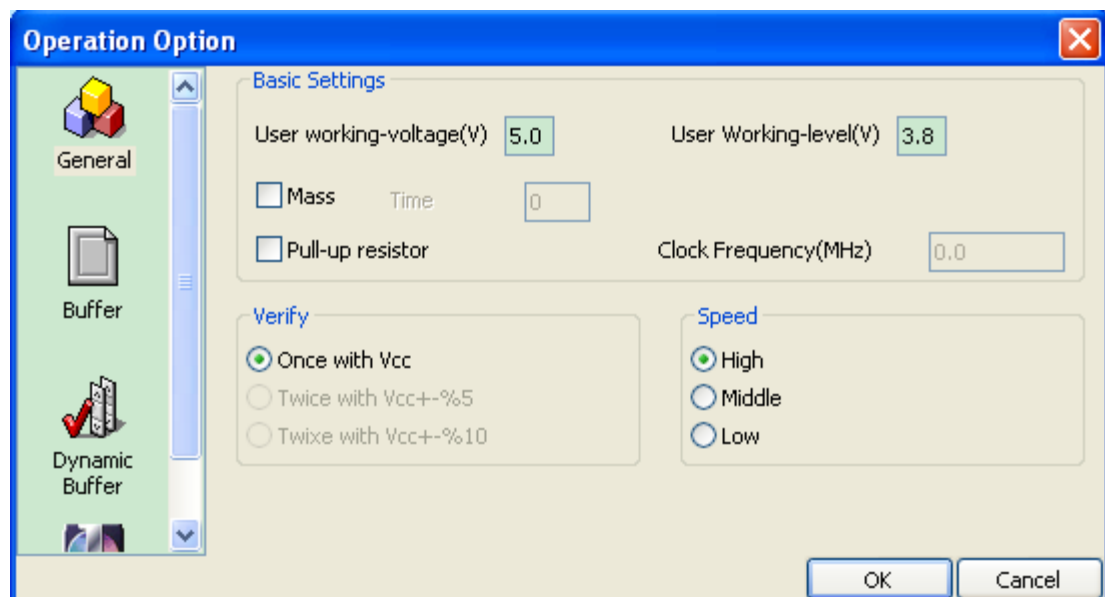
Delete: Remove selected function item from auto list (it also can be deleted by double-clicking the item).

Clear All: Delete all function items in the auto list.

## Operation options

Select “Operation option” in the device menu or  on the main toolbar, the following dialogue box pops up:

### General options



General Options include:

User's working voltage and working level

User sets working voltage and level according to the actual condition of target board.

Production mode

If production mode is selected, user should specify the delay time (required stabilization time after placing the chip).

The production mode is only valid under offline state.

Pull-up resistance

Clock frequency

Verify mode (only applicable to the case that the power of target board is supplied by the programmer)

After burning (programming) a chip, it is necessary to verify the burned data. According to the manufacturer's programming data, voltage applied to VCC pin has the following changes when making verification:



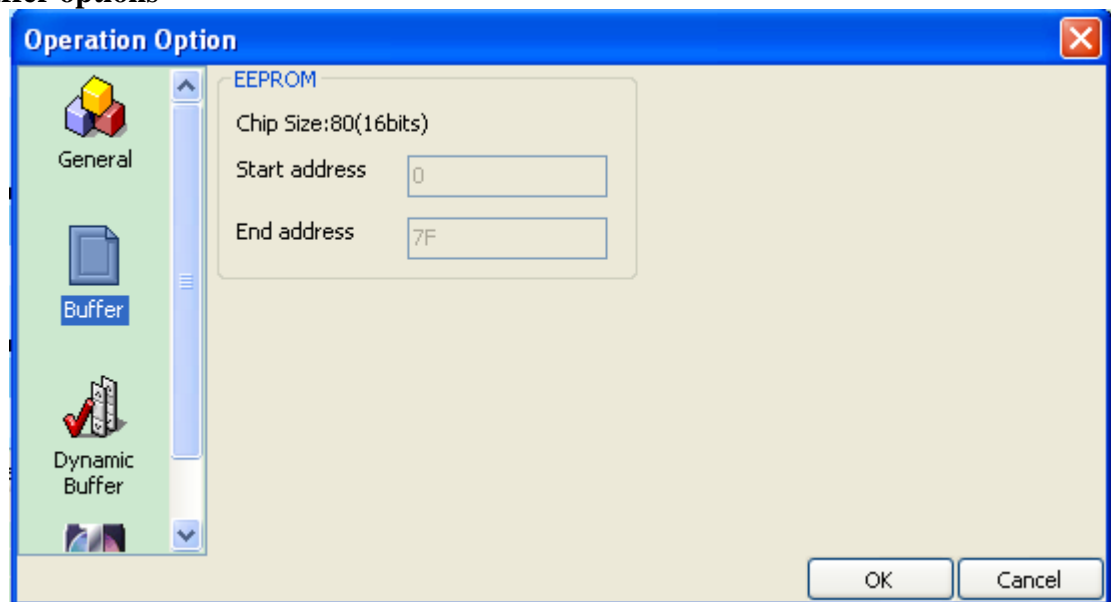
First, verify the data with  $VCC(\pm 5\%)$  or  $(\pm 10\%)$ , for example, for  $VCC=5.00V$ , then we can select  $VCC=5.00V$  to verify the data once, or use  $VCC=4.75V$  and  $VCC=5.25V$  to verify the data twice ( $\pm 5\%$ ) or use  $VCC=4.50V$  and  $VCC=5.50V$  to verify the data twice ( $\pm 10\%$ ).

Second, verify the data using the minimum verification voltage (MinVcc) and the maximum verification voltage (MaxVcc). The two voltage values referred above are supplied by the manufacturer.

### Speed

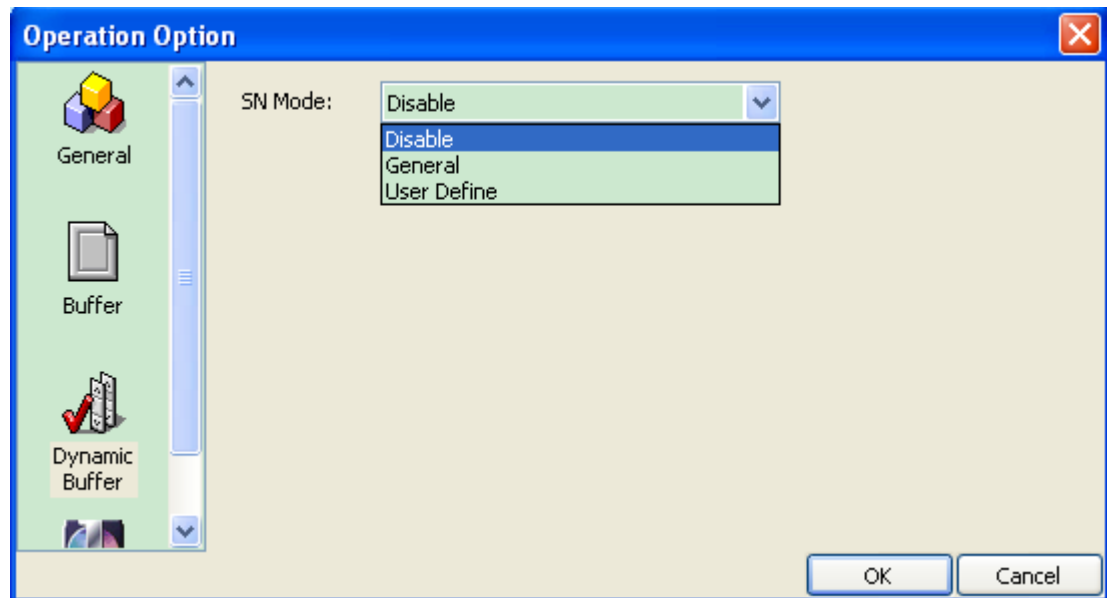
The user can decide the programming speed himself and can choose one of three speeds, i.e. high, medium and low (varies with chip).

### Buffer options

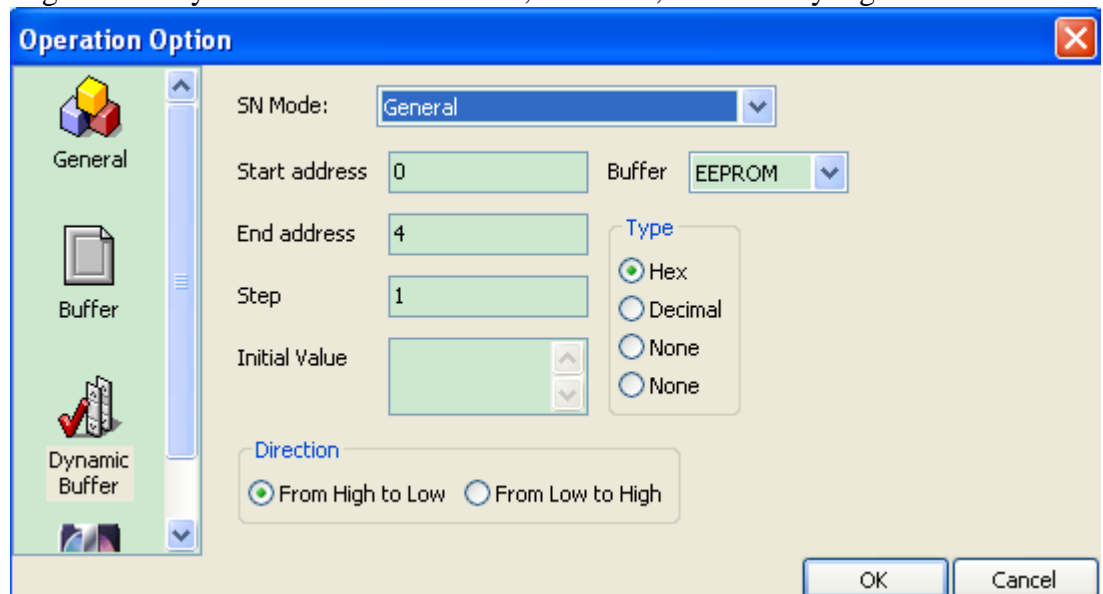


The user can select to burn a part of the chip, which is applicable to most part of E/EPROM (FLASH) devices.

### Serial number setting (dynamic buffer)



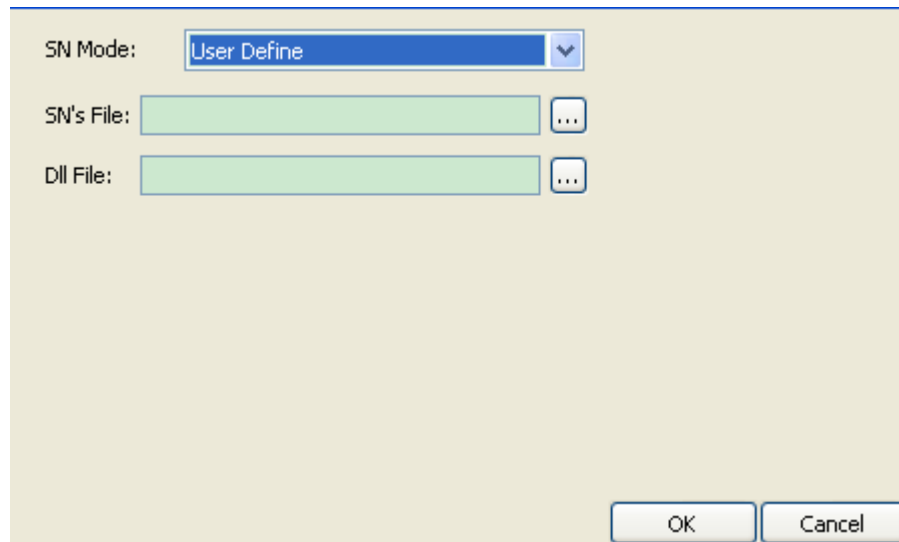
Some applications require writing different content to local areas of the chip, for example serial number of the product, MAC address etc., which are collectively called serial number. This software has provided two solutions. One is the standard increment method and the other is user-define method. The software for the former generates the serial number by automatic increment on the previous serial number. The format is fixed and it can meet general requirements. The serial number of the latter is generated by the user-define software, therefore, it has a very big freedom.



The user needs to fill the start and end address, step length and initial value. The user can select which buffer is to be incremented and how to increase (format, step and direction).

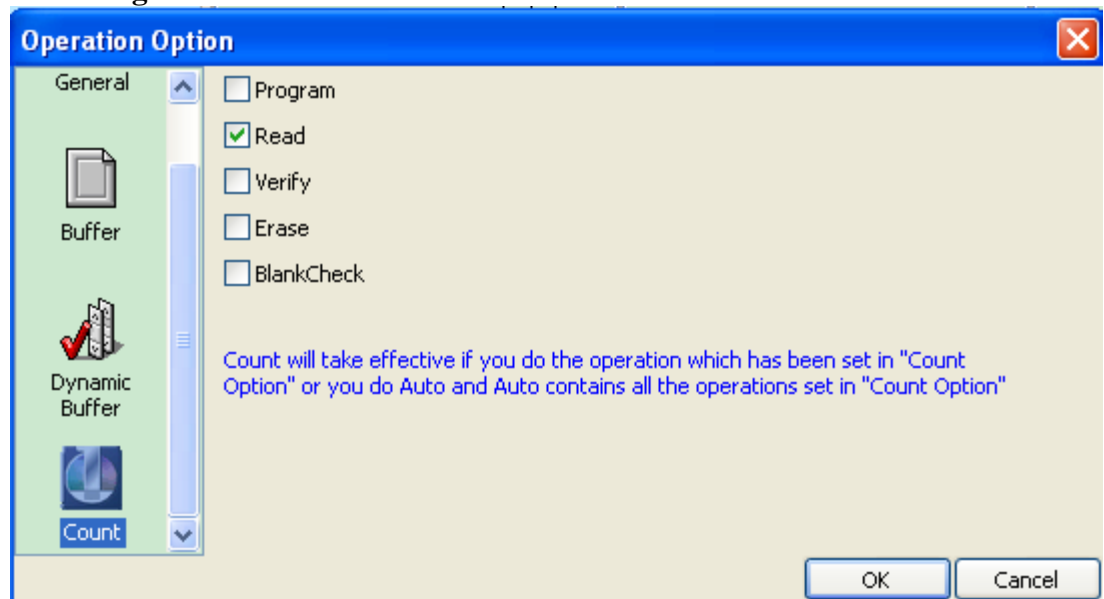
Note 1: What is reflected by the initial value is the current value, during burning process, if the user wants to change the serial number to be burned, he should change the initial value.

Note 2: the format of initial value must be correct, when the type is non-ASCII, the format of initial value is “XX XX XX”; when the type is ASCII, the format of initial value is “XXXXXX”. If the format is not correct, it will be considered as 0.



User-define mode. The user needs to provide the serial number file and dll called by the software, the serial number file needs to be generated correctly according to a certain format

### **Statistic setting**



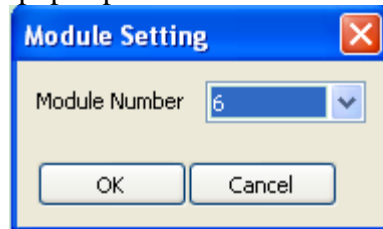
When burning the chip, the software will do statistics of programming according to user's setting.

Statistical method:

Count will take effective if you do the operation which has been set in "Count Option" or you do Auto and Auto contains all the operations set in "Count Option"

### **Module management**

Select “module management” on “options” menu bar, then the following dialogue box pops up:

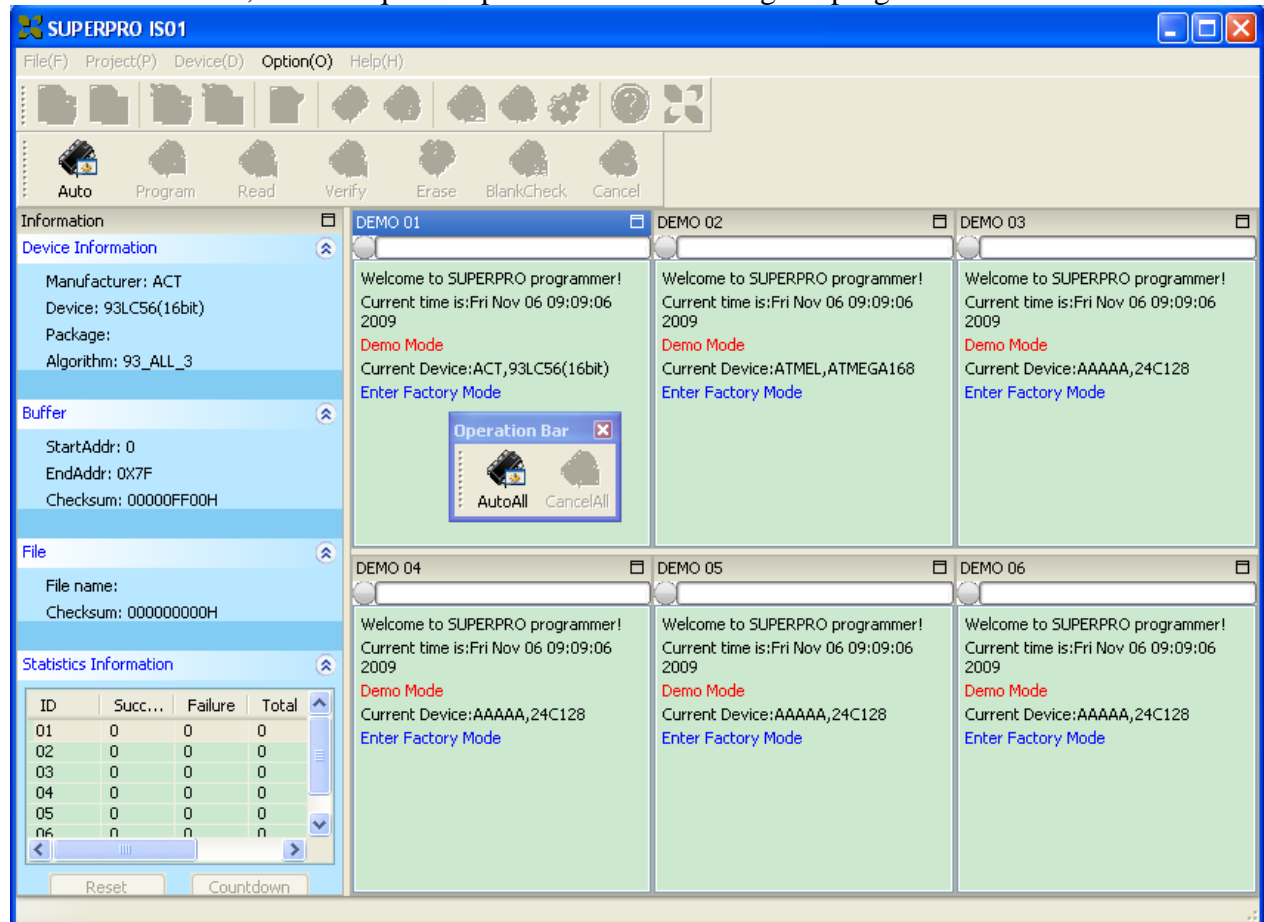


Note: The module setting only becomes valid after the software is started.

## Factory mode

It is designed for factory mass burning of chips. After entering the mode, the program will only allow executing limited functions such as "Auto" etc. in order to prevent the chip from being damaged by misoperation and prevent erroneous data from being burned.

For data security, the administrator can set a password when enter the mode. If password has been set, it also requires a password when exiting the program.



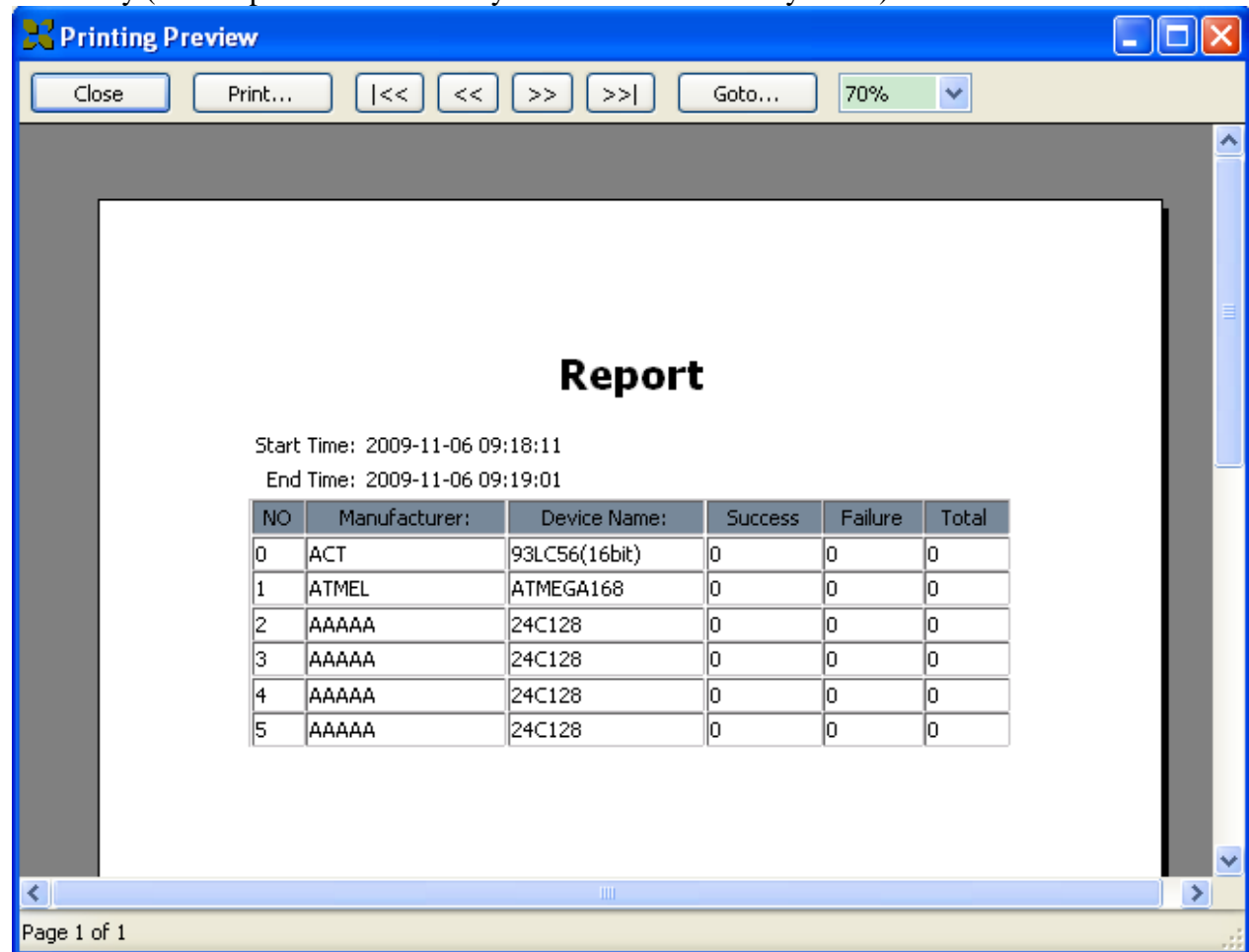
If the number of modules >1, there will appear the general operation



bar, click "AutoAll" to execute the respective Auto operations of each module; click "Cancel All" to cancel operations of all modules.

## Print Report

Choose "Print Report" menu on the menu bar, then the statistical report is generated automatically.(Print report function is only valid under the factory mode)



## Tool bar

### Main toolbar



- |                       |                              |                 |
|-----------------------|------------------------------|-----------------|
| 1. Open file          | 2. Save file                 | 3. Load project |
| 4. Save project       | 5. Edit buffer               | 6. Select chip  |
| 7. Device information | 8. Device Configuration Word | 9. Edit Auto    |
| 10. Operation Option  | 11. Help                     | 12. About XELEK |

## Operation toolbar

The items on operation toolbar vary with devices.  
Generally the operation bar of the chip is as follows:



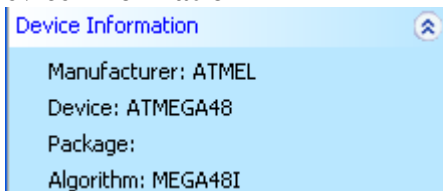
After selecting Xplayer, the operation toolbar is:




Select the operation to be carried out in drop-down list, then click “EXECUTE” to perform this operation or click “CANCEL” to stop.

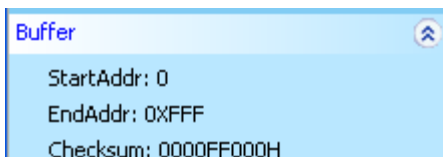
## Information bar

### Device information



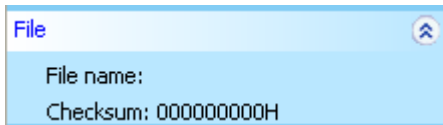
The device information bar shows the basic information of device, if the device has special information, you need click “Dev.Info” option on "devices" menu bar or  option on tool bar.

### Buffer information



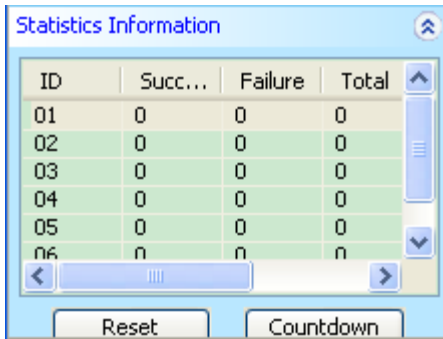
The buffer information bar shows start address, end address and checksum of main chip buffer.

### File information



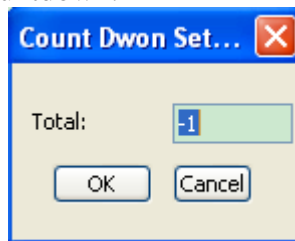
It shows the current loaded file name and file checksum (whether to calculate checksum or not is decided by user himself).

### Statistical Information Bar



Reset: Clear success and failure etc. to zero, and cancel countdown setting (countdown number is -1)





Countdown:



When the countdown reaches -1, it indicates the countdown setting has been cancelled.  
Note: The statistic mode needs to be set in "count" found in "operation option".

## Log window

### LED status display

-  It indicates LED is not connected to the programmer currently or the connection fails.
-  It indicates the programmer is connected successfully or operation is successful (programming verify etc.)
-  It indicates programmer operation fails (programming verify etc.)
-  It indicates the programmer is performing operation currently (programming verify etc.)

### Progress bar

It shows burning progress currently.



### Operation information bar

The information area where the programmer software interacts with the user, which shows operation progress, operation result and information historic records etc.

```
Welcome to SUPERPRO programmer!  
Current time is:Fri Nov 06 09:09:06 2009  
Demo Mode  
Current Device:ACT,93LC56(16bit)  
Enter Factory Mode  
Quit Factory Mode  
Current Device:ATMEL,ATMEGA48  
SUPERPRO IS01 Starts!  
SD Card Not Found.  
Current Device:AAAAA,24C32  
Batching...  
Writing...  
Operation failed!  
Address:000000  
Chip Data:0000  
Buffer Data:0000
```

## Offline operation

### Offline introduction

Turn on the system, the initialization isn't completed until the screen prompts "XELTEK IS01", then the system will wait for command. If the keyboard is triggered firstly, the system will enter offline mode (standalone work mode). Note that when the system enters offline mode, the system will first detect SD card, if it fails to detect the SD card, the system will prompt "NO SD FOUND!" on the display screen, at this time please insert SD card and reboot the system.

### Outline

It is operated through local keyboard and LCD display without connecting with PC. SD card stores project algorithms and user data files set by him. The user can create his own project in online mode and download it into the card, he can also copy data files directly into SD card through a card reader, meanwhile, notice that the project suffix must be .prj, otherwise the system will not consider it as a project file.

All contents relevant to user's offline operations are stored in it in the form of "PROJECT" file which includes the device type, data file, device configuration bit setting etc. In short, when booting the machine, the user can complete all settings just by selecting "project" name in the menu, then the machine starts working. The project must be created online and downloaded to the SD card, which can simplify offline operation and avoid production losses due to misoperation. The number of projects stored by SD card at one time is only limited by card capacity. When creating and downloading project library, if the library size exceeds the capacity of SD card, the software will give a remind, at this time, the user should reduce the number of projects or insert a SD card of larger capacity. For specific



method of project construction and download, please refer to previous sections in the operation manual.

## Menu description

### Keyboard rules:

ENTER—select and enter next-level menu

EXIT—exit to the upper level menu

↑, ↓ scroll up and down select. BUFFER—change the address where the cursor lies on the display

→, ← BUFFER—select address bit on the display

### 1. RUN

Finish the actual operation of device under this menu. The actual operational functions of devices vary slightly; the common functions are as follows:

AUTO—perform user-defined AUTO function sequence (Set at the time of project construction)

PROGRAM—write the data in BUFFER into chip

READ—read the data in the chip into SD card and save it as data0.

BLANK CHECK—check whether the chip is empty

VERIFY—Compare whether content of written chip is consistent with that in BUFFER

ERASE—Electrically erase the chip content

SECURE—Encrypt the chip with security function

LOCK/MEMORY PROTECT etc. have the same function with this function

If you do not select a project, the system will prompt “NO PRJ SELECTED”, please select a project first.

### 2. SELECT PROJECT

Select a project stored in SD card by user. Use ↑ or ↓ key to select an appropriate project after entering. After pressing ENTER again, the algorithm and data of selected project are invoked automatically. After selecting a model each time, selected result will be automatically saved so that the user needn't to select it again when he starts the machine and can directly enter RUN (but if the project has a password, you need to re-select the model). If the project has a password, you can use →, ← key to change the position and ↑, ↓ key to change the password digit on current position in the password input window.

### 3. SETTING

1) READ OPTION (Switch with read function)

Operation is same as above. Inhibit as a default.

### 4. BUFFER

#### 1) CHECK SUM

Calculate the checksum of data files in the project and compare it with CHECKSUM previously stored and calculated in current project, if there is a difference, CHECKSUM values are displayed, respectively. Both physical and man-made causes are likely to result in contamination or damage of SD card data, which results in scrapping of the burned chip. Therefore, it is suggested to check the CHECKSUM at least once each time before starting the system.

#### 2) DISPLAY BUFFER

BUFFER content is displayed. At the position of DISPLAY BUFFER, press ENTER to display the first address and data. Press → or ← to move the cursor to the position of address which you want to modify.

Press ↑ or ↓ to change the address where the cursor lies. Press ENTER, 1 will be added to the address in turn.

## **5. MASS PRODUCTION**

There are two production modes. Refer to the content as described above.

## **Appendices**

### **License Agreement**

The copyright of the program and User's Guide remain the property of XELTEK.

#### **You may:**

Copy the program for back-up purposes ONLY in support of its use on a single computer.

Transfer the program and license to another party if the other party agrees to accept the terms and conditions of this agreement.

#### **You may not**

Use this product in a computer system or network, which allows the program to be operated by more

than one user at a time.

Modify, copy, or transfer the User's Guide, other documentation or any copy.

Reverse engineering, de-compiling, or disassemble any program module or security device.

### **Limited Warranty**

XELTEK warrants that its products will be free from defects in workmanship and materials, and shall

conform to specifications current at the time of shipment, for a period of one year from the date of

shipment. During the one-year period, any defective software or hardware products will be repaired or

replaced, at seller's discretion, on a return to factory basis.

This warranty applies only to products properly installed and operated within specified environmental conditions.

#### **XELTEK's responsibility under this warranty does not apply to:**

Any product that has been repaired, worked upon, or altered by any person not duly authorized in writing

by XELTEK.

Any product that has been subject to misuse, negligence, accident, or its serial number has been altered.

Any program error caused by physically damaged CD.

The extended and joint liability of damages caused by hardware faults or software bugs.

### **Customer Support**

SUPERPRO software is user friendly for your usage and requires minimal technical support. The program comes with a comprehensive User's Guide. If you cannot find the answer in the manual, you

can turn to your dealer or distributor or XELTEK. XELTEK provides telephone (number available at

website) technical assistance during normal business hours from 9:00am to 5:00pm. Please have your invoice number ready when calling, as we cannot answer your questions without it. Software updates are available free from the web site.

**Pre-call Check List:**

If you are having difficulty understanding the program, have you studied the User's Guide?

If there seems to be a problem with the software, can you reproduce it?

If the program has displayed an error message, please write down the message.

When phoning or faxing or e-mailing for technical assistance, have your invoice number ready.

Are you using the newest version of the software?

# AMEYA360

Components Supply Platform

Authorized Distribution Brand :



Website :

Welcome to visit [www.ameya360.com](http://www.ameya360.com)

Contact Us :

➤ Address :

401 Building No.5, JiuGe Business Center, Lane 2301, Yishan Rd  
Minhang District, Shanghai , China

➤ Sales :

Direct     +86 (21) 6401-6692  
  
Email       amall@ameya360.com  
  
QQ         800077892  
  
Skype      ameyasales1 ameyasales2

➤ Customer Service :

Email       service@ameya360.com

➤ Partnership :

Tel         +86 (21) 64016692-8333  
  
Email      mkt@ameya360.com